Permutation Puzzles: A Mathematical Perspective

15 Puzzle, Oval Track, Rubik's Cube and Other Mathematical Toys

Lecture Notes

Jamie Mulholland
Department of Mathematics
Simon Fraser University

Contents

Preface

These are the notes for the course *Math 302 - Mathematics of Permutations Puzzles*. The aim of the course is to provide an introduction to group theory for students in the mathematics minor program. Group theory is a very powerful branch of mathematics, but is often difficult for one to enter into due to its very abstract nature. Puzzles like Rubik's cube can provide concrete models for some of these abstractions.

The header of each page contains an image 📄 which takes you right back to the table of contents.

A number of online resources are available on the website accompanying these notes [10]:

http://www.sfu.ca/ jtmulhol/math302/

No project such as this can be free from errors and incompleteness. I will be grateful to everyone who points out any typos, incorrect statements, or sends any other suggestion on how to improve this manuscript.

Jamie Mulholland
Simon Fraser University
j_mulholland@sfu.ca
June 4, 2013

Greek Alphabet

lower case	capital	name	pronunciation	lower case	capital	name	pronunciation
α	A	alpha	(al-fah)	ν	N	nu	(new)
β	B	beta	(bay-tah)	ξ	Ξ	xi	(zie)
γ	Γ	gamma	(gam-ah)	o	O	omicron	(om-e-cron)
δ	Δ	delta	(del-ta)	π	Π	pi	(pie)
ε	E	epsilon	(ep-si-lon)	ρ	P	rho	(roe)
ζ	Z	zeta	(zay-tah)	σ	Σ	sigma	(sig-mah)
η	H	eta	(ay-tah)	τ	T	tau	(taw)
θ	Θ	theta	(thay-tah)	υ	Υ	upsilon	(up-si-lon)
ι	I	iota	(eye-o-tah)	ϕ	Φ	phi	(fie)
κ	K	kappa	(cap-pah)	χ	X	chi	(kie)
λ	Λ	lambda	(lamb-dah)	ψ	Ψ	psi	(si)
μ	M	mu	(mew)	ω	Ω	omega	(oh-may-gah)

Lecture 1

Permutation Puzzles

1.1 Introduction

Imagine a mixed up Rubik's Cube$^{\text{TM}}$ (for example see Figure 1.1a), or better yet, mix up your own cube. As you begin to try to solve the cube you should notice a few things. Solving a single face (i.e. getting all 9 pieces of the same colour onto the same face) isn't that difficult. There seems to be enough room to move things around. Continuing in this manner, you then begin to solve the next layer. You'll soon notice that certain moves will undo your previous work. If you twist a face that contains some pieces that you had previously put in their correct place, then these pieces now move out of place. And this is where the puzzle becomes challenging. It seems the more pieces that are in the correct place, the harder it is to move the remaining ones into place. This is known as the *end-game* of the Rubik's Cube, and solving the puzzle requires a thorough understanding of this part of the puzzle.

(a) A mixed up Rubik's Cube

(b) An end-game for Rubik's Cube. Only the bottom layer remains to be solved.

Figure 1.1: A mixed up Rubik's Cube and end-game.

Rubik's Cube is probably one of the most well known puzzles to date. It is estimated that over 350 million have been sold since it's creation by Ernö Rubik around 1980. What has made it so popular is not certain. Perhaps it looks seemingly innocent, then once a few sides have been twisted, and the colours begin to mix, the path back home is not so easy to see. The more you twist it the further you seem to be taken away from the solution. Perhaps it is that the number of ways to mix up the cube seems endless. Or perhaps to others, it doesn't seem endless at all. Despite the reasons for its appeal, it has become one of the most popular puzzles in history.

It is rare to find a puzzle, or toy, that has captured the imagination of millions, is accessible to all age levels, is challenging, yet satisfying, and is so *mathematically rich*. The Rubik's Cube is one such puzzle. Others examples include the 15-puzzle, TopSpin, Hungarian Rings, and Lights Out.

What do we mean by mathematically rich? Well it turns out that one area of mathematics that has had an impact on all areas of science, and has even popped up in art, is the area called *group theory*. Often referred to as the *language of symmetry*, group theory has led to many new discoveries in theoretical physics, chemistry, and mathematics itself. It underlies the techniques in cryptography (sending private information over public channels), and coding theory (digital communications, digital storage and retrieval of information). It is no surprise that a mathematical theory developed to understand symmetry so adequately describes the Rubik's Cube, however, for us, we are more interested in the opposite. We will use Rubik's Cube, and these various other puzzles, to provide us a window into group theory. But most of all, we plan to have a lot of fun doing it.

Our goal in this course is to uncover some pretty fascinating mathematics while playing with puzzles. We will not be too concerned with solving the puzzles, though strategies for solution will fall out of our investigations, but instead we want to see how we can model these puzzles mathematically and see what the mathematics has to tell us about the puzzles. In this sense we want to *understand* these puzzles.

This is the theme for all these puzzles. There is a certain stage in solving the puzzle where a simple strategy, and trial and error, can't get you any further. This is typically referred to as the *end-game* for the puzzle. For Rubik's Cube the end-game occurs when two layers are solved, and the last layer remains (see Figure 13.1c).

It is understanding the *end-game* of these puzzles that mathematics becomes such a useful tool.

1.2 A Collection of Puzzles

We begin by briefly describing the puzzles we will be investigating in this course. One thing to observe is that all puzzles have a common theme: the pieces of the puzzle are rearranged, and the goal is to return the pieces to their proper (original) arrangement.

1.2.1 A basic game, let's call it *Swap*

Imagine a set of objects laid out in front of you and ordered in some way. This puzzle can be played with any number of objects, but the more objects that are used the more challenging it becomes.

It doesn't matter what the objects are, they could all be different, or some could be the same. For starters we will just use 5 distinct objects, and for simplicity we will just take the objects to be the numbers 1, 2, 3, 4, and 5. Figure 1.3 shows the objects laid out in front of us:

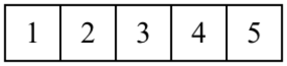


Figure 1.2: *Solved state* of *Swap* with 5 objects.

This arrangement, where the numbers appear in order from left to right, is called the *home position* or *solved state*. Since, as we'll see shortly, we will be moving the numbers around the boxes so it will be nice to have a little reminder of whose home is whose. We do this by putting a little number in the top left corner of each box.

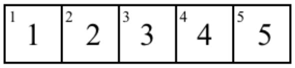
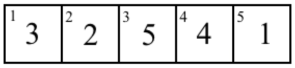


Figure 1.3: *Solved state* of *Swap* with 5 objects with the boxes labeled.

The way this puzzle is played is first the numbers are randomly arranged in the boxes. Then using only *legal moves*, one tries to move the numbers back to their home positions (i.e. return the puzzle to its solved state).

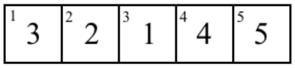
What are the *legal moves*? This is where different version of the puzzle can be created. For now, let us simply say there is one type of legal move called a *swap*, and it consists of picking any two boxes and swapping the contents (the numbers in large font).

Example 1.1: Consider the starting position shown in Figure 1.4. Our goal is to return the numbers to the solved state using only legal moves. Notice that objects 2 and 4 are in the correct positions (correct boxes). This is where the little numbers in the top left corners come in handy. As for the numbers 1, 3 and 5, we need to move these to their correct positions. Since the legal moves consist of swapping the contents of two boxes at a time, we'll focus first on getting 5 into it's correct position. To do this we swap the contents of boxes 3 and 5, since object 5 is in box 3.



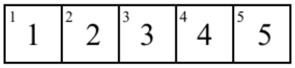
1	2	3	4	5
3	2	5	4	1

Figure 1.4: Staring position for Example 1.1.



1	2	3	4	5
3	2	1	4	5

Now 2, 4 and 5 are in the correct positions. Lastly we swap the contents of boxes 1 and 3 and solve the puzzle.



1	2	3	4	5
1	2	3	4	5

■ [1]

This is a pretty basic puzzle, but it is good to have this as our starter puzzle. In a certain sense, as we'll soon see, every puzzle we will investigate will just be some variation of Swap. Either we will increase the number of objects, or we will change the *legal moves*. We now consider some possible variations of the puzzle.

Variations of Swap: There are many ways to vary this puzzle. One way is to increase the number of objects that are used. Here we used 5, but we could use 10, 20, 48, or any number we wish.

Another way to vary the puzzle is to choose a different collection of legal moves. Our legal moves consisted of swapping the contents of two boxes. Instead we could have stated that legal moves only consist of swapping the contents of box 1 and any other box. If this was the case then the solution in Example 1.1 would be illegal, since it started by swapping the contents of boxes 3 and 5, which is a move that doesn't involve box 1.

Exercise 1.1: Beginning with the starting position in Figure 1.4, solve the puzzle using only legal moves of the form: the contents of any box can only be swapped with box 1. In other words, any swap must involve box 1.

[1] The black square symbol ■ is used to indicate the example is finished. Later, when we prove theorems, lemmas, etc. we will use a hollow square □ to denote the end of a proof.

We could also extend the notion of a legal move beyond ”swaps”. For instance we could restrict ourselves to use only moves of the form: *pick three boxes and cycle the contents either to the right (clockwise) or to the left (counterclockwise)*.

For example, consider 6 objects in Figure 1.5, we could cycle the contents of boxes 2, 3 and 5 to the left (other boxes are shaded to allow us to focus on what is changing).

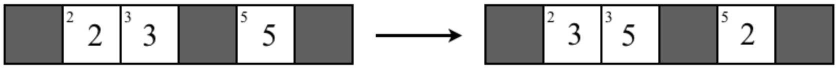


Figure 1.5: Legal move variation: 3-cycle to the left.

Exercise 1.2: Beginning with the starting position in Figure 1.4, solve the puzzle using only legal moves consisting of 3-cycles: pick three boxes and cycle the contents either to the right (clockwise) or to the left (counterclockwise).

We can now describe the puzzle **Swap**, including all possible variations.

Rules of Swap (n objects):

Let T be a set of n objects, and suppose the objects have been ordered in some way. For example, the objects can be the numbers 1 to n and the ordering could be their natural ordering from smallest to largest written from left to right. When the objects are in their proper order, we say they are in their *home positions* and the puzzle is in the *solved state*. Let $\mathcal{M}$ be a collection of *legal moves*.

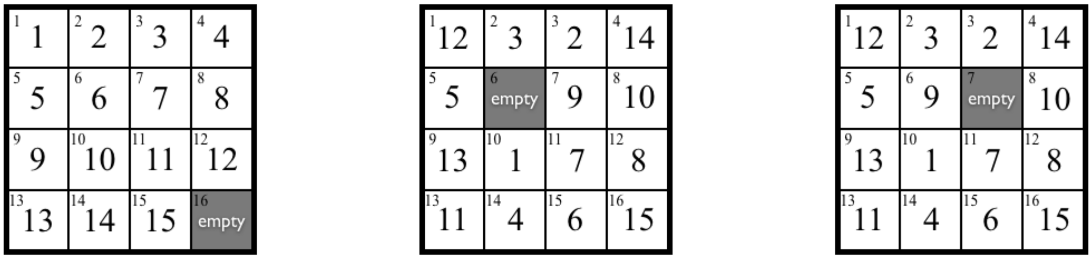
(a) **Puzzle Start**: Randomly arrange the numbers 1 through n from left to right.

(b) **Puzzle Play**: Using only legal moves (i.e. moves in $\mathcal{M}$), return the puzzle to the solved state.

Stated here in its most general form we’ll see that most Rubik’s Cube-like puzzles are just variations of Swap. Of course, this connection with Swap doesn’t make the Rubik’s Cube any easier to solve, at least not yet, but it will provide us a way to investigate and understand the cube and other puzzles.

1.2.2 The 15-Puzzle

The 15-puzzle consists of a 4×4 grid with numbered tiles from 1 to 15 placed in the grid. The space where the 16 tile would go is left empty. See Figure 1.6a.



(a) The 15-Puzzle in the solved state

(b) A random arrangement of the 15-Puzzle.

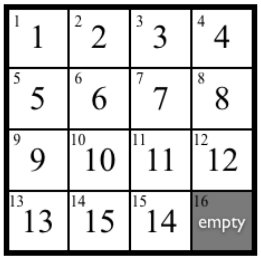
(c) Obtained from 1.6b by moving the tile in box 7 (tile number 9) to box 6.

Figure 1.6: The 15 Puzzle

The little numbers in the top left corner of each box are not present on any of the manufactured puzzles, nor are they present on the software versions of the puzzle. But these little numbers proved to be so handy in reminding us where each tiles home position is in Swap that we'll use them here to.

The object of this puzzle is to randomly arrange the tiles on the grid, and then through a sequence of legal moves which consist of sliding a tile into the empty space (which results in the empty space moving around the board), one tries to return all tiles to their home positions.

Most currently manufactured versions of the puzzle consist of a tongue-and-groove design which allows the pieces to slide around but doesn't allow them to be removed. However, the original versions of the puzzle (manufactured in the 1880's) consisted of removable wooden pieces. This little difference in the construction has significant impact on the solvability of the puzzle. This puzzle started a craze that swept across the nation, and across the world, from January to April of 1880. All the fuss was centred around the fact that after randomly putting the wooden blocks back into the box, solving the puzzle seemed to take you to one of two places: either you solved it completely, or you got every number in its correct position except the 14 and 15 were switched (see Figure 1.7). In the case when the last two tiles were switched it seemed the puzzle wasn't solvable. Cash rewards of $1000 were offered for a solution, and one dentist even offered a set of teeth to the person who could produce a sequence of moves swapping the 14 and 15 tiles.



^{1} 1	^{2} 2	^{3} 3	^{4} 4
^{5} 5	^{6} 6	^{7} 7	^{8} 8
^{9} 9	^{10} 10	^{11} 11	^{12} 12
^{13} 13	^{14} 15	^{15} 14	^{16} empty

Figure 1.7: The 13-15-14 Problem. Can the puzzle be solved by starting from this position?

We will investigate whether this arrangement of the puzzle is solvable, as well as come up with a strategy for solving the puzzle in general.

Software: This puzzle is widely available as a free download for various operating system (mac/ win/ linux/ ios/ android). Most versions have some sort of picture as the background, instead of the numbers 1 through 15. To find out more see [10].

1.2.3 The Oval Track Puzzle (or TopSpin$^{\text{TM}}$)

The TopSpin puzzle was manufactured by Binary Arts (now called ThinkFun). It was invented by Ferdinand Lammertink, and patented on 3 Oct 1989, US 4,871,173. The puzzle consists of 20 numbered round pieces in one long looped track (see Figure 1.8). You can slide all the pieces around the loop. There is also a turntable in the loop (this is the purple circle which contains disks 1 through 4 in Figure 1.8), which can rotate any four adjacent pieces so that they will be in reverse order. This swaps two adjacent pieces and the two pieces on either side of them. The aim of this puzzle is to mix up the ordering of the pieces, and then place the pieces back in numerical order (as shown in Figure 1.8).

This puzzle became a North American classic with over a million copies sold. Nowadays the most common place to find this puzzle in its physical form is on ebay.

Variations of Oval Track: The name *Oval Track* has been given to a more general version of the puzzle, one that is updated for the digital era. By considering virtual versions of the puzzle one can disregard the mechanical constraints imposed by physical construction, this opens up a whole new world of possibilities

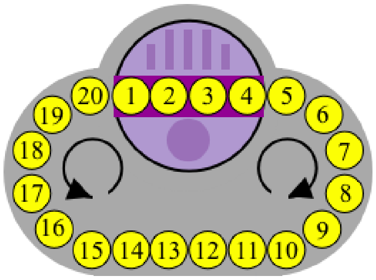
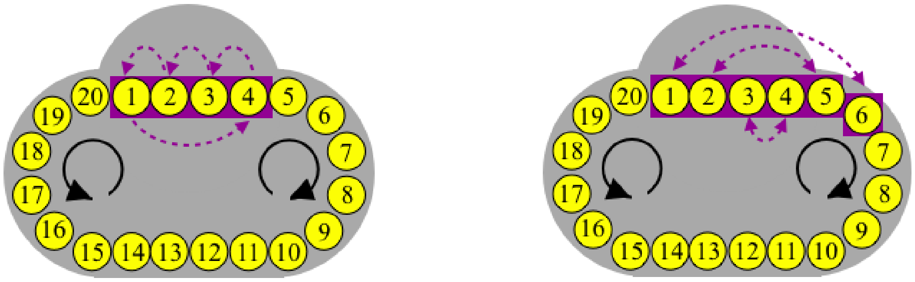


Figure 1.8: The TopSpin Puzzle in its solved state.

for the turntable move. For example, Figure 1.9 shows two variations of the puzzle. The *turntable move* in the original TopSpin puzzle is now replaced with the move indicated by the purple dashed lines. The new *turntable move* for the puzzle in Figure 1.9a moves the disk in spot 4 to spot 3, the disk in spot 3 to spot 2, the disk in spot 2 to spot 1, and takes the disk in spot 1 to spot 4. Another version of the *turntable move* involving 6 disks is given in Figure 1.9b. Variations of this puzzle are now limited only by your imagination.



(a) One variation of the turntable move

(b) Another variation of the turntable move

Figure 1.9: Variations on the Oval Track Puzzle

As usual, it will be convenient to indicate the home positions of the disks. We put little numbers around the track indicating the number of the disk that should be in that position in the solved puzzle. See Figure 1.10.

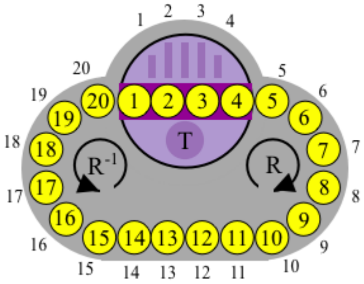


Figure 1.10: The TopSpin Puzzle with home positions labeled, and move notation.

Oval Track Notation: A clockwise rotation of numbers around the track, where each number moves one space, is denoted by R. A counterclockwise rotation is denoted by R^{-1}. A rotation of the turntable is (in general, an application of the *follow the arrows* move) is denoted by T. Figure 1.10 provides a visual summary of this notation.

Software: To find a virtual version of this puzzle see [10].

1.2.4 Hungarian Rings

The Hungarian Rings puzzle consists of two intersecting rings made up of a number of coloured balls. The rings of balls intersect at two places, so they share two of the balls. Each ring of balls can be rotated, so the balls can be mixed. The aim is to mix up the balls, and then place the balls back together so the colour form a continuous sequence (as show in Figure 1.11).

There are 38 balls of four colours: two colours have 9 balls (yellow and blue) and two colours have 10 balls (black, red). There are 4 balls between the intersections of the rings.

In the Rubik's Cubic Compendium [page 212], there is a picture of the Hungarian Rings and the following text by David Singmaster:

> Closer to Rubik's Magic Cube are 'interlocking cycle' puzzles where several rings of pieces cross each other. Endre Pap, a Hungarian engineer, invented a flat version with two rings which was marketed as the Hungarian Rings. The idea was not entirely new, as there is an 1893 patent for it.

The patent that Singmaster is referring to is US 507,215 by William Churchill, filed on May 28 1891, granted on October 24, 1893. For a copy of the patent see Jaap's Puzzle Page [13].

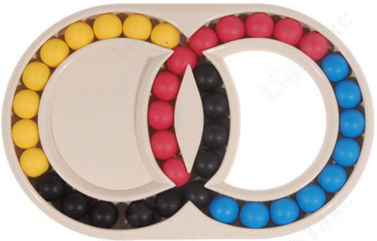
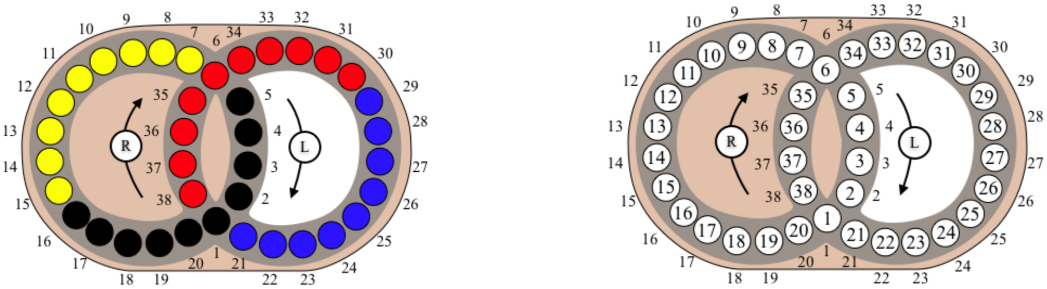


Figure 1.11: Hungarian Rings in its solved state. (manufactured 1982)

To study this puzzle we will temporarily ignore colours, and instead assign a number to each ball. See Figure 1.12b. We'll also indicate the home position of each ball by putting little numbers along the outside of the track. In effect we will study the puzzle of rearranging the numbers 1 through 38 on the two rings. In some sense this is a more difficult puzzle than the colour version of the puzzle simply because in the colour version there are really only 4 distinct balls, whereas in the number version there are 38 distinct balls and each one has only one home position. However, as we'll see, the added complexity inherited by using numbers as labels, rather than colours, is manageable, and the benefits gained in understanding the puzzle are numerous.

Hungarian Rings Notation: A clockwise rotation of the balls in the **right-hand ring**, where each ball moves one space around the track, is denoted by R, a counterclockwise rotation is denoted by R^{-1}. A clockwise rotation of the balls in the **left-hand ring**, where each ball moves one space around the track, is denoted by L, a counterclockwise rotation is denoted by L^{-1}. Figure 1.12 provides a visual summary of this notation.

Software: To find a virtual version of this puzzle see [10].



(a) Hungarian Rings with home positions labeled by numbers.

(b) Hungarian Rings with numbers instead of colours.

Figure 1.12: Hungarian Rings with disks labeled by numbers

1.2.5 Rubik's Cube

Rubik's Cube is probably the most well known mechanical puzzle. It was invented by Ernő Rubik in Hungary around 1974, and the patent was filed 30 January 1975, HU 170062. Eventually it was produced and marketed by Ideal Toys in the early 1980s. It is quite possibly the most popular toy to have ever been manufactured, and many copycat cubes were made. It is estimated that there have been over 350 million cubes manufactured since 1980, and it is still being manufactured today.

Rubik's Cube is a cube which is built from smaller cubes where there are 3 cubes along an edge, i.e. a $3 \times 3 \times 3$ cube. The 9 pieces on each face can rotate, which rearranges the small cubes at that face. The six sides of the puzzle are coloured, so every corner piece shows three colours, every edge piece shows two colours, and every face centre only one. See Figure 1.13.

(a) View of front (red), right (yellow) and up (blue) faces.

(b) View of back (orange), left (white) and down (green) faces.

Figure 1.13: The $3 \times 3 \times 3$ Rubik's Cube with classic colouring scheme: blue opposite green, red opposite orange, white opposite yellow.

Turning a face does not change the face centres so these can be considered already solved. (This is because twisting the face centres is not a visible change of pattern. However, if there was an image rather than a solid colour on the face then this would not be the case anymore.) The other pieces have to be placed correctly around the centres. This is a particularly important observation because it implies the following:

> The colour of the centre piece of any face defines the only colour to which that face of the cube can be restored.

Cube Terminology & Notation: When playing with the cube the pieces begin to move all around. Since there are so many moving parts of the cube, it will be convenient to have some terminology to describe each piece, and its placement in the cube. It will also be convenient to have some notation for basic movements to aid in communication with one another. Of course, a good choice of notation can bring mathematics into the picture as well, as we will soon see. The notation we use was first introduced by David Singmaster in the early 1980's, and is the most popular notation in use today.

Fix an orientation of the cube in space. We may label the 6 sides as ***f***, ***b***, ***r***, ***l***, ***u***, ***d*** for *front*, *back*, *right*, *left*, *up*, and *down*.

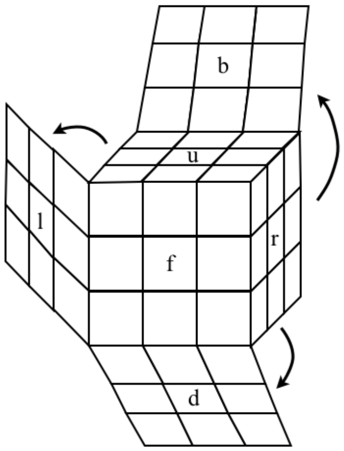
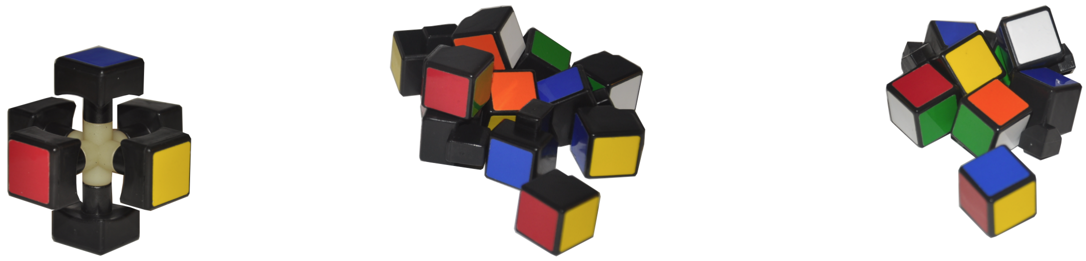


Figure 1.14: Labeling sides of Rubik's Cube by f, b, r, l, u, d.

The cube is made up of 26 smaller cubes called **cubies**. These are the ones that are visible, there actually isn't a 27th cube in the middle, but instead a mechanism that allows things to twist and turn. The cube has 6 sides, or **faces**, each of which has $3 \cdot 3 = 9$ **facets**. Think of a facet as the position a little coloured sticker can occupy. There are 54 facets in total for the $3 \times 3 \times 3$ Rubik's Cube. The cubes split up into three types: **centre cubies** (having only one sticker), **edge cubies** (having two stickers), **corner cubies** (having three stickers). See Figure 1.15.



(a) 6 centre cubies and mechanism

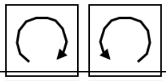
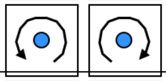
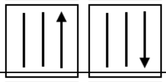
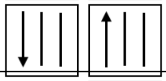
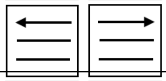
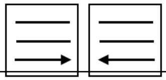
(b) 12 edge cubies

(c) 8 corner cubies

Figure 1.15: A disassembled Rubik's Cube showing the cubies.

Each face of the cube is made up of a slice of 9 cubies that share a facet with the face. The face, along with all of the 9 cubies in the slice, can be rotated by 90 degrees clockwise (viewing the the face straight-on). We denote this move by an uppercase letter of the name of the face. For example, F denote the move which rotates the front face by 90 degrees clockwise. See Table 1.1 for a complete description of cube moves and notation.

We call the space that a cubie can occupy a **cubicle**. As the pieces move around, the cubies move from cubicle to cubicle, and the coloured stickers move from facet to facet. In order to solve the puzzle each cubie must get restored to its original cubicle, we call this its **home location**, and each coloured sticker must

notation	pictorial	description of basic move
(Singmaster)	**(view from front)**	**(clockwise/counterclockwise refers to viewing the face straight-on)**
F , F^{-1}		F = quarter turn of **front** face in the **clockwise** direction. F^{-1}= quarter turn of **front** face in the **counterclockwise** direction.
B , B^{-1}		B = quarter turn of **back** face in the **clockwise** direction. B^{-1}= quarter turn of **back** face in the **counterclockwise** direction.
R , R^{-1}		R = quarter turn of **right** face in the **clockwise** direction. R^{-1}= quarter turn of **right** face in the **counterclockwise** direction.
L , L^{-1}		L = quarter turn of **left** face in the **clockwise** direction. L^{-1}= quarter turn of **left** face in the **counterclockwise** direction.
U , U^{-1}		U = quarter turn of **up** face in the **clockwise** direction. U^{-1}= quarter turn of **up** face in the **counterclockwise** direction.
D , D^{-1}		D = quarter turn of **down** face in the **clockwise** direction. D^{-1}= quarter turn of **down** face in the **counterclockwise** direction.

$F^2, B^2, R^2, L^2, U^2, D^2$ denote the corresponding *half-turn* of the face.
Since a clockwise half-turn is equivalent to a counterclockwise half-turn then
$F^2 = F^{-2}$, $B^2 = B^{-2}$, $R^2 = R^{-2}$, $L^2 = L^{-2}$, $U2 = U^{-2}$, $D^2 = D^{-2}$

Table 1.1: Summary of cube move notation

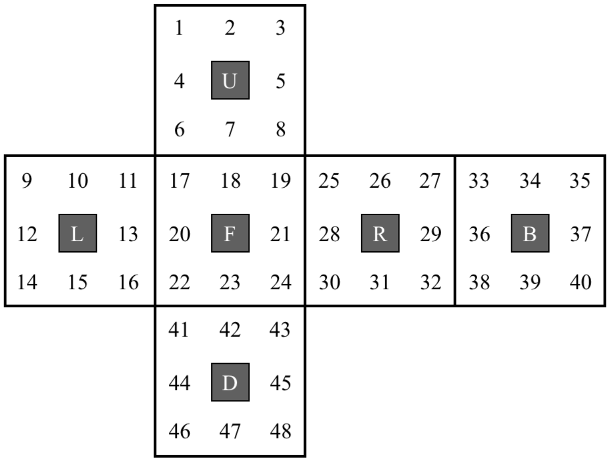
get restored to its original facet, we call this the cubies **home orientation**. Once *all* cubies are in their home positions and home orientations the puzzle will be solved.

Table 1.2 summarizes the terminology introduced here.

Terminology	**Definition or Abbreviation**
cubies	The small cube pieces which make up the whole cube.
sticker	A colored face of a cubie.
cubicles	The spaces occupied by the cubies.
facets	The faces of a cubicle, or space occupied by a sticker.
types of cubies: **corner**, **edge**, and **centre**:	A corner cubie has three facets. An edge cubie has two facets. A centre cubie has one facet
home location - of a cubie	The cubicle to which a cubie should be restored.
home position - of a cubie	The orientation in the home location to which a cubie should be restored.
positional names for cube faces	Up (u) Down (d) Right (r) Left (l) Front (f) Back (b)
Notation for cubicles - shown in *italics*	Lower case initials. For example, *uf* denotes the Up-Front edge cubicle, *dbl* denotes the Down-Back-Left cubicle.
Notation for cubies - shown in *italics*	Upper case initials. For example, *URF* denotes the cubie whose home position is in the the Up-Right-Front corner (i.e. cubicle *ufr*).

Table 1.2: Summary of terminology and notation

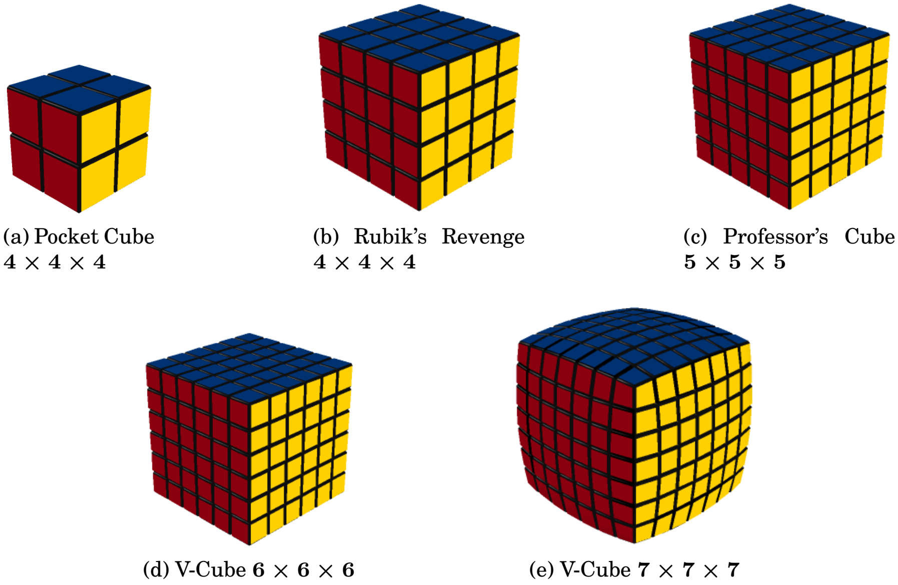
Since the center facets are fixed by the basic moves there are only $54 - 6 = 48$ facets that move. If we label the facets as in Figure 1.16 then we see that each basic move corresponds to a rearrangement, or *permutation* of the numbers 1 through 48. In this way we see that the Rubik's Cube is much like the puzzle Swap, in the sense that we have a set of 48 numbers and a set of legal moves $\mathcal{M}$ (the 6 basic cube moves) which allow us to rearrange these numbers in some way.



			1	2	3						
			4	U	5						
			6	7	8						
9	10	11	17	18	19	25	26	27	33	34	35
12	L	13	20	F	21	28	R	29	36	B	37
14	15	16	22	23	24	30	31	32	38	39	40
			41	42	43						
			44	D	45						
			46	47	48						

Figure 1.16: Facet labeling on the $3 \times 3 \times 3$ Rubik's Cube.

Variations of Rubik's Cube: The Rubik's Cube is the puzzle that started the whole *twisty puzzle* craze. Since its invention hundreds of different types of twisty puzzle of all shapes and sizes have been designed. Puzzles of this type are often called *Rubik's Cube-like puzzles*, or *twisty puzzles*, or *permutation puzzles*. Figure 1.17 gives some examples.



(a) Pocket Cube $\mathbf{4 \times 4 \times 4}$

(b) Rubik's Revenge $\mathbf{4 \times 4 \times 4}$

(c) Professor's Cube $\mathbf{5 \times 5 \times 5}$

(d) V-Cube $\mathbf{6 \times 6 \times 6}$

(e) V-Cube $\mathbf{7 \times 7 \times 7}$

Figure 1.17: $n \times n \times n$ Cubes

1.3 Which brings us to the Definition of a Permutation Puzzle

The 15-Puzzle, Oval Track puzzle, Hungarian Rings and Rubik's Cube are all variations of the same theme. Each one consisted of pieces that were rearranged, or permuted, in some way, and the goal is to try to restore the pieces to their original positions. The legal moves that one is allowed to use is forced by the design or construction of the puzzle. Puzzles of this type known as *permutation puzzles*. Since these type of puzzles are the main focus of this course, we shall give a precise definition for this term.

A **one person game** is a sequence of moves following certain rules which satisfy:

- there are finitely many moves at each stage,
- there is a finite sequence of moves which yields a solution,
- there are no "chance" or "random" moves (such as rolling a dice to determine what to do next),
- there is complete information about each move,
- each move depends only on the present position, not on the existence or non-existence of a certain previous move (such as chess, where castling is made illegal if the king has been moved previously).

A **permutation puzzle** is a one person game (solitaire) with a finite set $T = \{1, 2, \ldots, n\}$ of puzzle pieces satisfying the following four properties:

(a) For some $n > 1$ depending only on the puzzle's construction, each move of the puzzle corresponds to a unique permutation of the numbers in T,

(b) If the permutation of T in (a) corresponds to more than one puzzle move then the two positions reached by those two respective moves must be indistinguishable,

(c) Each move, say M, must be "invertible" in the sense that there must exist another move, say M^{-1}, which restores the puzzle to the position it was at before M was performed, In this sense, we must be able to "undo" moves.

(d) If M_1 is a move corresponding to a permutation τ_1 of T and if M_2 is a move corresponding to a permutation τ_2 of T then $M_1 \cdot M_2$, which denotes the move M_1 followed by the move M_2 (notice the order the moves are applied is from left to right), is either

- not a legal move, or
- corresponds to the permutation $\tau_1\tau_2$.

Notation: We will always write successive puzzle moves from *left to right*, as we did in step (d) above.

1.4 Exercises

1. Get your own Rubik's Cube. Whether you buy or borrow, make sure you have access to a Rubik's Cube while working through this book. If you don't know where to buy one, then check the course website, some suggestions are posted.

2. Get familiar with Rubik's Cube, and all the other puzzles we have just discussed. The website accompanying this book has links to virtual versions of the puzzles: http://www.sfu.ca/ jtmulhol/math302/. Download your own copy of the ones that are available, and bookmark the ones that are "online only" versions. Play with these puzzles. Don't worry if you can't solve them, this will come. But for now just get familiar with the puzzles, and the movements of the pieces.

3. Solve the Swap puzzle given in Figure 1.18, using the original set of legal move: *swap the contents of any two boxes*.

1	2	3	4	5	6
2	6	4	1	3	5

Figure 1.18: Swap position for Exercises 3, 4, 5.

4. Solve the Swap puzzle in Figure 1.18, using only legal moves of the form: *the contents of any box can only be swapped with box* 1.

5. Can you solve the Swap puzzle given in Figure 1.18, using only legal moves consisting of 3-cycles: *pick three boxes and cycle the contents either to the right (clockwise) or to the left (counterclockwise)*?

6. Consider the starting arrangement of tiles for the Swap puzzle in Figure 1.19.

   (a) Solve the puzzle using only legal moves of the form: *the contents of any box can only be swapped with box* 1.

   (b) Solve the puzzle using only legal moves consisting of 3-cycles: *pick three boxes and cycle the contents either to the right (clockwise) or to the left (counterclockwise)*.

   (c) Solve the puzzle using only legal moves consisting of pairs of swaps: *pick four boxes, swap the contents of two boxes, and swap the contents of the other two boxes.*

1	2	3	4	5	6
1	6	4	2	3	5

Figure 1.19: Swap position for Exercise 6.

7. Can you solve the Swap puzzle given in Figure 1.18, using only legal moves consisting of pairs of swaps: *pick four boxes, swap the contents of two boxes, and swap the contents of the other two boxes*?

8. Verify that each of the puzzles we've encountered: Swap, 15-Puzzle, Oval Track, Hungarian Rings and Rubik's Cube, are permutation puzzles. That is, show that the definition of the term "permutation puzzle" is satisfied by these puzzles.

Lecture 2

A Bit of Set Theory

2.1 Introduction

Rubik's Cube is made up a number of different pieces: corner cubies, edge cubies, and center cubies (see Lecture 1 for definitions of these terms). Each collection of these pieces forms a set. In order to understand how these pieces move around we need to understand how the cube moves F, B, R, L, U, D act on these sets. In this lecture we recall some basic terminology and notation from set theory which will form the foundation for our mathematical investigations into Rubik's Cube and other puzzles.

2.2 Sets and Subsets

A **set** is a well-defined collection of objects. The objects of the set are called **elements**, and are said to be **members** of, or **belonging** to, the set.

By *well-defined* we mean that for any element we wish to consider, we are able to determine, under some scrutiny, whether or not it is a member of the set.

Typically we will use capital letters, such as A, B, C, ... to represent sets and lower case letters to represent elements. For a set A we write $x \in A$ **if** x **is an element of** A, and $y \notin A$ **if** y **is not an element in** A.

Sets are usually defined in one of two ways:

(a) Listing all of its elements in braces: $A = \{a, b, c, \ldots\}$. For example $A = \{1, 2, 3, 4, 5\}$ is the set of integers from 1 to 5. Therefore, $3 \in A$, but $6 \notin A$.

(b) Using *set-builder* notation: $A = \{x \mid x \text{ has property } P\}$. For example we could define the previous set A as $\{x \mid x \text{ is an integer and } 1 \leq x \leq 5\}$. The vertical bar " | " is read "such that" . The symbols $\{x \mid \ldots\}$ are read "the set of all x such that ...". Some authors use a colon ":" instead of " | ", so we could also write $A = \{x : x \text{ has property } P\}$.

Some basic sets of numbers we should be familiar with are:

- $\mathbb{Z} =$ the set of *integers* $= \{\ldots, -2, -1, 0, 1, 2, 3, \ldots\}$,
- $\mathbb{N} =$ the set of *nonnegative integers* or *natural numbers* $= \{0, 1, 2, 3, \ldots\} = \{x \in \mathbb{Z} \mid x \geq 0\}$,
- $\mathbb{Z}^+ =$ the set of *positive integers* $= \{1, 2, 3, \ldots\} = \{x \in \mathbb{Z} \mid x > 0\}$,

- $\mathbb{Q} =$ the set of *rational numbers* $= \left\{ \frac{a}{b} \,\middle|\, a, b \in \mathbb{Z}, b \neq 0 \right\}$,

- $\mathbb{Q}^+ =$ the set of *positive rational numbers* $= \{x \in \mathbb{Q} \mid x > 0\}$,

- $\mathbb{R}$ is the set of *real numbers*.

- $[n] = \{1, 2, \ldots, n\}$ = the set of integers from 1 to n, where $n \in \mathbb{Z}^+$.

Let A and B be sets. If all the elements of A also belong to B then we say A is a **subset** of B and we write $\boldsymbol{A \subset B}$. For example, $\mathbb{Z}^+ \subset \mathbb{Z}$ since every positive integer is itself an integer, but $\mathbb{Q} \not\subset \mathbb{Z}$, since there are rational numbers that are not integers, for example $\frac{1}{2}$.

Two sets A and B are said to be **equal**, and we write $\boldsymbol{A = B}$, if $A \subset B$ and $B \subset A$.

If a set has a finite number of elements then we say it is a **finite set**. Otherwise it is an **infinite set**. For any finite set A, $|A|$ denotes the number of elements in A and is called the **cardinality**, or *size*, of A. For example, $|[n]| = n$, whereas $\mathbb{Z}$ in an infinite set.

The **empty set**, or **null set**, is the set that contains no elements. The empty set is denoted by $\emptyset$, or $\{\}$, and has that property that $|\emptyset| = 0$.

Let A and B be two sets. The set of all elements belonging to either A or B is denoted $\boldsymbol{A \cup B}$ and is called the **union** of A and B. The set of all elements belonging to both A and B is denoted $\boldsymbol{A \cap B}$ and is called the **intersection** of A and B. The set of all elements not belonging to A is denoted A^c or sometimes by $\overline{A}$, and is called the **complement** of A. [1] The set of all elements in A that are not in B is denoted $\boldsymbol{A - B}$ and is called the **difference of** A **with** B. We sometimes refer to this as A **take away** B.

The **Cartesian product** of A and B is the set of all ordered pairs (x, y) where $x \in A$ and $y \in B$ and is denoted by $\boldsymbol{A \times B}$.

The following summarizes the different operations we have on sets:

$$\begin{aligned}
A \cup B &= \{x \mid x \in A \textbf{ or } x \in B\}, \\
A \cap B &= \{x \mid x \in A \textbf{ and } x \in B\}, \\
A^c = \overline{A} &= \{x \mid x \notin A\}, \\
A - B &= \{x \mid x \in A \textbf{ and } x \notin B\} = A \cap B^c \\
A \times B &= \{(x, y) \mid x \in A \text{ and } y \in B\}.
\end{aligned}$$

We call two sets **disjoint** if they have no element in common: A and B are disjoint if $A \cap B = \emptyset$.

2.3 Laws of Set Theory

Some of the major laws that govern set theory are the following.

For any sets A, B, and C taken from a universe $\mathcal{U}$

[1] In defining the complement we need to specify the elements we are considering. That is, we need to consider A as a subset of some larger set. To see why, just think about what could be meant by $\mathbb{Z}^c$? Does this mean all elements in $\mathbb{Q}$ not in $\mathbb{Z}$, or all elements in $\mathbb{R}$ not in $\mathbb{Z}$, or something else entirely. The larger set to which we consider A as a subset will be called the *universe* or *universe of discourse* denoted by $\mathcal{U}$. It will be clear, given the context, as to what universe we are working in. What this means though is that we should really write $A^c = \{x \mid x \in \mathcal{U} \text{ and } \notin A\}$.

1)	$(A^c)^c = A$	Law of Double Negation
2)	$(A \cup B)^c = A^c \cap B^c$	DeMorgan's Laws
	$(A \cap B)^c = A^c \cup B^c$	
3)	$A \cup B = B \cup A$	Commutative Laws
	$A \cap B = B \cap A$	
4)	$A \cup (B \cup C) = (A \cup B) \cup C$	Associative Laws
	$A \cap (B \cap C) = (A \cap B) \cap C$	
5)	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	Distributive Laws
	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	
6)	$A \cup A = A$	Idempotent Laws
	$A \cap A = A$	
7)	$A \cup \emptyset = A$	Identity Laws
	$A \cap \mathcal{U} = A$	
8)	$A \cup A^c = \mathcal{U}$	Inverse Laws
	$A \cap A^c = \emptyset$	
9)	$A \cup \mathcal{U} = A$	Domination Laws
	$A \cap \emptyset = \emptyset$	
10)	$A \cup (A \cap B) = A$	Absorbtion Laws
	$A \cap (A \cup B) = A$	

These set theoretic laws are similar to the arithmetic properties of the real numbers, where "$\cup$" plays the role of "$+$", and "$\cap$" plays the role of "$\times$". However, there are several differences.

We will prove the first part of the Distributive Law, and leave the proof of all others to the reader. See Exercise 8 and 9 for the second part of the Distributive Law and DeMorgan's Law. Also see Exercise 10 where the reader is asked to provide proofs for the remaining laws of set theory.

Proof: Let $x \in \mathcal{U}$. Then

$$\begin{aligned} x \in A \cup (B \cap C) \quad &\Leftrightarrow \quad x \in A \quad \text{or} \quad x \in B \cap C \\ &\Leftrightarrow \quad x \in A \quad \text{or} \quad x \text{ is in both } B \text{ and } C \\ &\Leftrightarrow \quad x \in A \cup B \quad \text{and} \quad x \in A \cup C \\ &\Leftrightarrow \quad x \in (A \cup B) \cap (A \cup C) \end{aligned}$$

This completes the proof. □

We also state a result about the cardinality of a disjoint union of sets.

Theorem 2.1: *Let $A_1, A_2, \ldots, A_n$ be disjoint finite sets. Then*

$$|A_1 \cup \cdots \cup A_n| = |A_1| + \cdots + |A_n|.$$

Proof: In the case of two sets (i.e. $n = 2$), let A and B be two disjoint finite sets and write $A = \{a_1, a_2, \ldots, a_k\}$ and $B = \{b_1, b_2, \ldots, b_\ell\}$. Since A and B are disjoint then

$$A \cup B = \{a_1, a_2, \ldots, a_k, b_1, b_2, \ldots, b_\ell\}$$

and so $|A \cup B| = k + \ell = |A| + |B|$. The general case follows by mathematical induction. □

2.4 Examples Using Sage

Example 2.1: In this example we show how to define a set, and compute cardinalities, unions, intersections, differences and cartesian products using Sage.

Sage

```
sage: S1=Set([1,2,3,4,5]);
sage: S2=Set([3,4,5,6,7]);
sage: S1;S2;
{1, 2, 3, 4, 5}
{3, 4, 5, 6, 7}
sage: S1.cardinality()
5
sage: S1.union(S2)
{1, 2, 3, 4, 5, 6, 7}
sage: S1.intersection(S2)
{3, 4, 5}
sage: S1.difference(S2)
{1,2}
sage: S2.difference(S1)
{6,7}
sage: CartesianProduct(S1, S2)
Cartesian product of {1, 2, 3, 4, 5}, {3, 4, 5, 6, 7}
sage:  CartesianProduct(S1,S2).list()
[[1, 3], [1, 4], [1, 5], [1, 6], [1, 7], [2, 3], [2, 4], [2, 5], [2, 6],
[2, 7], [3, 3], [3, 4], [3, 5], [3, 6], [3, 7], [4, 3], [4, 4], [4, 5],
[4, 6], [4, 7], [5, 3], [5, 4], [5, 5], [5, 6], [5, 7]]
sage: CartesianProduct(S1,S2).cardinality()
25
sage: 2 in S1
True
sage: 1 in S2
False
```

Example 2.2: Sage has a number of commonly used sets already built in: $\mathbb{Z}$, $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{R}$. The commands are ZZ, NN, QQ, and RR, respectively.

Sage

```
sage: ZZ
Integer Ring
sage: 1 in ZZ
True
sage: 1/2 in ZZ
False
sage: 0 in NN
True
sage: -1 in NN
False
```

Example 2.3: We can build a set by using properties in Sage. Here we use Python's modulo operator %: $a\%b$ returns the remainder of a when divided by b.

Sage

```
sage: Set(x for x in range(0,10) if x%2==0)
{0, 2, 4, 6, 8}
sage: Set(x for x in range(0,10) if x%2==1)
{1, 3, 9, 5, 7}
```

Example 2.4: The `is_prime()` function returns True if the input is a prime integer, and False if not. Such functions are called **boolean valued functions**. We can use boolean valued function to create subsets as this example shows.

Sage

```
sage: is_prime(29)
True
sage: is_prime(4)
False
sage: Set(x for x in range(0,10) if is_prime(x))
{2, 3, 5, 7}
sage: Set(x for x in range(0,1000) if is_prime(x)).cardinality()
168
```

The last computation shows there are 168 prime numbers less than 1000.

We can also use the `filter()` command in Python. You can get more information on `filter()` by typing `filter?` at the Sage prompt.

Sage

```
sage: S=Set(1..20)  #constructs a set of all integers from 1 to 20
sage: filter(is_prime,S)
[2, 3, 5, 7, 11, 13, 17, 19]
```

2.5 Exercises

1. Which of the following sets are equal?

   (a) $\{1, 2, 3\}$ (b) $\{2, 3, 1, 3\}$ (c) $\{3, 2, 1, 1, 2\}$ (d) $\{1, 3, 3, 2, 1, 3\}$

2. Let $A = \{1, \{1\}, \{2\}\}$. Which of the following statements are true?

   (a) $1 \in A$ (c) $\{1\} \subset A$ (e) $\{2\} \in A$ (g) $\{\{2\}\} \subset A$
   (b) $\{1\} \in A$ (d) $\{\{1\}\} \subset A$ (f) $\{2\} \subset A$ (h) $\{1, 2\} \subset A$

3. Determine all the elements of the following sets.

   (a) $\{1 + (-1)^n \mid n \in \mathbb{N}\}$
   (b) $\{n \in \mathbb{N} \mid n \leq 20$ and n is divisible by $3\}$
   (c) $\{n \in \mathbb{N} \mid n \leq 20,\ n$ is prime, and $2n + 1$ is divisible by $3\}$

4. Determine the cardinality of the following sets.

   (a) The set of all cubies of the Rubik's Cube which have a blue facet.
   (b) The set of all corner cubies of the Rubik's Cube which have a blue facet.

5. Consider the set $A = \{1, 2, 3, 4, 5\}$.

   (a) How many subsets of cardinality 1 does A have?
   (b) How many subsets of cardinality 2 does A have?

(c) How many subsets does A have in total?
(Hint: don't forget the empty subset, and the set A itself, when counting subsets.)

6. For $\mathcal{U} = [10]$, let $A = \{1, 2, 3, 4, 5\}$, $B = \{1, 2, 4, 8\}$ and $C = \{1, 2, 3, 5, 7\}$. Determine each of the following.

   (a) $(A \cup B) \cap C$

   (b) $(A \cup B) - C$

   (c) $A^c \cap B^c$

   (d) $|A \cup B|$

7. Verify your answers to question 6 by using Sage.

8. Prove the second *Distributive Law* stated in Section 2.3.

9. Prove *DeMorgan's laws* stated in Section 2.3.

10. Prove all the remaining laws of set theory that are stated in Section 2.3.

Lecture 3

Permutations

The puzzles we have encountered so far all have a common theme: the pieces can be mixed up, and the goal is to restore the pieces back to some proper order. In this lecture we will introduce some terminology and notation for talking about rearrangements of objects. In particular, we give the definition of a *permutation*, which is the main object we will use to study puzzles. We'll also discuss *permutation multiplication*, *inverses*, and *order*.

3.1 Permutation: Preliminary Definition

In mathematics, the notion of *permutation* is used with several slightly different meanings, all related to the act of permuting (rearranging in an ordered fashion) objects or values. Informally, a permutation of a set of objects is an arrangement of those objects into a particular order.

Example 3.1: There are six permutations of the objects in the set $\{\clubsuit, \diamondsuit, \heartsuit\}$, namely $[\clubsuit, \diamondsuit, \heartsuit]$, $[\clubsuit, \heartsuit, \diamondsuit]$, $[\diamondsuit, \clubsuit, \heartsuit]$, $[\diamondsuit, \heartsuit, \clubsuit]$, $[\heartsuit, \clubsuit, \diamondsuit]$, and $[\heartsuit, \diamondsuit, \clubsuit]$.

Notation: Curly braces $\{,\}$ denote *sets*, i.e. the order that elements are listed doesn't matter. Square braces $[,]$ denote lists, i.e. the order that elements appear does matter. So as sets $\{1,2,3\} = \{2,1,3\}$ but as lists $[1,2,3] \neq [2,1,3]$.

Sage

```
sage: Set([1,2,3])==Set([2,1,3])
True
sage: [1,2,3]==[2,1,3]
False
```

Example 3.2: There are 5040 ways to arrange the seven books in the Harry Potter series on your bookshelf. If we let 1 denote Volume 1: Philosopher's Stone, 2 denote Volume 2: Chamber of Secrets, etc. then, for example, two possible permutaions are $[1,3,5,7,2,4,6]$ and $[5,2,1,3,7,4,6]$. Of course, out of all these possible permutations the most likely way to place them on the bookshelf is in numerical order: $[1,2,3,4,5,6,7]$.

To determine the number of permutations we imagine 7 empty slots on the bookshelf which we are about to fill. There are 7 ways to pick a book and place it in slot 1. For each of these choices, there are now 6

possible ways to fill slot 2, then 5 possible ways to fill slot 3, etc. So the total number of ways to fill the 7 slots is: $7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 7! = 5040$.

Sage

```
sage: factorial(7)
5040
```

Example 3.3: In the game of Swap on 5 objects the empty puzzle board is shown in Figure 3.1a.

1
2
3
4
5
1 3
2 2
3 5
4 4
5 1

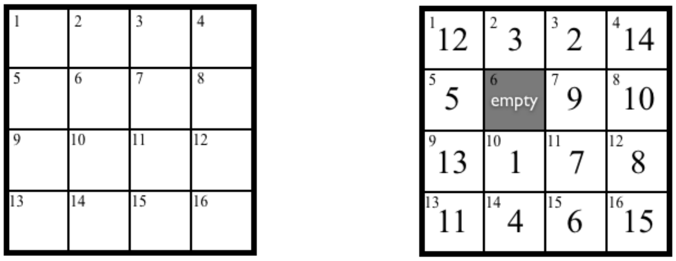
(a) The empty Swap board

(b) A random arrangement of Swap.

Figure 3.1: Game of Swap

The puzzle board is filled by laying out the tiles numbered 1 through 5 in the boxes. For example, one such puzzle position is shown in Figure 3.1b. Each puzzle position corresponds to a permutation of the set $[5] = \{1, 2, 3, 4, 5\}$. There are $5! = 120$ permutations of $[5]$, and so there are 120 different possible positions in the game of Swap. Only one of which is the solved state.

Example 3.4: The fifteen puzzle with no tiles in the boxes is shown in Figure 3.2a.



(a) The empty 15-Puzzle board

(b) A random arrangement of the 15-Puzzle.

Figure 3.2: The 15 Puzzle

The puzzle is started by placing the 15 tiles anywhere on the board. For example, one such puzzle position is shown in Figure 3.2b. This corresponds to a permutation of the set $[16] = \{1, 2, \ldots, 16\}$, where we imagine the blank space as being the 16th tile. There are $16! = 20,922,789,888,000$ permutations of $[16]$, and so there are $16!$ different ways to lay the tiles on the board. As for which positions are actually solvable, this is one of the key questions we will investigate later.

We can use Sage to generate permutations of a list, for example $[1, 2, 3]$.

Sage

```
sage: terms=[1,2,3];
sage: Permutations(terms)
Permutations of the set [1, 2, 3]
sage: Permutations(terms).list();
[[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]]
sage: number_of_permutations(terms)
6
sage: factorial(3)
6
```

Sometimes when listing permutations of a set we will omit the square braces. For example the 6 permutations of $[1, 2, 3]$ can be listed as: $123, 132, 213, 231, 312, 321$.

We can also list permutations of a multi-set, that is a set with more than one element repeated. Though, to define a multi-set we would actually need to use a list.

Example 3.5: Two permutations of the multi-set $[a, a, b, b, b]$ are $[b, a, b, a, b]$ and $[b, b, a, a, b]$. There are $\dfrac{5!}{2! \cdot 3!} = 10$ permutations in total. (Since there are $5!$ ways to arrange 5 objects, but 2 of the objects are identical, and so are the other 3.)

Sage

```
sage: var('a,b');
sage: terms=[a,a,b,b,b];
sage: Permutations(terms)
Permutations of the multi-set [a, a, b, b, b]
sage: Permutations(terms).list();
[[a, a, b, b, b], [a, b, a, b, b], [a, b, b, a, b], [a, b, b, b, a], [b,
a, a, b, b], [b, a, b, a, b], [b, a, b, b, a], [b, b, a, a, b], [b, b,
a, b, a], [b, b, b, a, a]]
sage: number_of_permutations(terms)
10
sage: factorial(5)/(factorial(2)*factorial(3))
10
```

3.2 Permutation: Mathematical Definition

It will be convenient for us to have a slightly more mathematical definition of a *permutation*. Before we give this formal definition however it is best to start by recalling the notion of a *function*, and the properties: *one-to-one*, and *onto*.

3.2.1 Functions

Definition 3.1: A **function**, or **mapping**, f from a (nonempty) set A to a (nonempty) set B is a rule that associates each element $a \in A$ to exactly one element $b \in B$.

Notation & Terminology: We write $f : A \to B$ to denote a function named f from set A to set B. A is called the **domain** of f and B the **codomain** . If f sends a to b then we write $f(a) = b$, or $f : a \mapsto b$. We also say b is the **image** of a under f. The subset of B consisting of all images $f(a)$, for $a \in A$, is called the **range** of f, and is written:

$$f(A) = \{f(a) \mid a \in A\} \subset B.$$

See Figure 3.3 for a pictorial representation of these ideas.

Definition 3.2: A function $f : A \to B$ is called **one-to-one**, or **injective**, if each element of B appears at most once as the image of an element of A.
A function $f : A \to B$ is called **onto**, or **surjective**, if $f(A) = B$. That is, if each element of B is the image of at least one element of A.
A function that is both injective and surjective is called **bijective** .

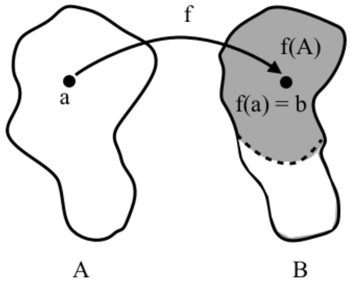


Figure 3.3: The way to visualize a function f, domain A, codomain B and range (shaded region).

3.2.2 Permutations

We are now ready to give the definition of a permutation.

Definition 3.3: A **permutation** of a set A is a function $\alpha : A \to A$ that is bijective (i.e. both one-to-one and onto).

Our goal is to understand how the pieces of a puzzle move around, so we typically represent each piece by a number, that is by an element of $[n] = \{1, 2, 3, \dots, n\}$. A rearrangement of the pieces then corresponds to a bijection from $[n] \to [n]$, in other words a *permutation* as defined above.

Unlike in calculus, where most functions are defined on infinite sets and given by formulas, permutations of finite sets are usually given by simply listing where each value goes.

For example, we can define a permutation α of the set $\{1, 2, 3\}$ by stating:

$$\alpha(1) = 2, \quad \alpha(2) = 1, \quad \alpha(3) = 3.$$

In Sage we can use the `Permutation()` command to construct a permutation. Here we define the permutation by the list of images $[\alpha(1), \alpha(2), \dots]$.

Sage

```
sage: a=Permutation([2,1,3]); a
[2,1,3]
sage: a(1)
2
sage: a(2)
1
```

A slightly more convenient way to represent this permutation is by:

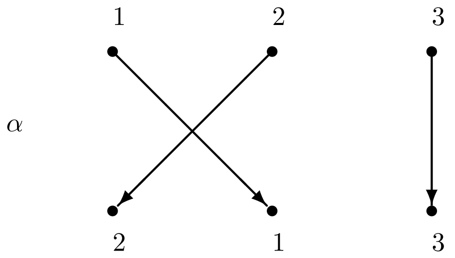
$$\alpha \leftrightarrow \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$$

where the top row are the element of $[3] = \{1, 2, 3\}$ and the bottom row are the corresponding images under α. This is known as *array notation* for a permutation.

Here is an example of how to use matrices in Sage to display a permutation in array form. One way is to use the `matrix()` command, where the syntax is
`matrix( [ <list for row 1> , <list for row 2> ] ).`

```
sage: a=Permutation([2,1,3])
sage: matrix([[1,2,3],[a(i) for i in [1,2,3]]]);
[1 2 3]
[2 1 3]
```

A more visual representation is by means of an *arrow diagram*. The arrows point from x to $\alpha(x)$.



Array Notation: In general, we may define a permutation $\alpha : [n] \to [n]$ by a $2 \times n$ array:

$$\alpha \leftrightarrow \left(\begin{array}{cccc} 1 & 2 & \dots & n \\ \alpha(1) & \alpha(2) & \dots & \alpha(n) \end{array} \right).$$

Since α is bijective the second row would just be a rearrangement of the numbers in the top row.

Example 3.6:

(a) The **identity** permutation, denoted by ε, or I, is the permutation that does nothing:

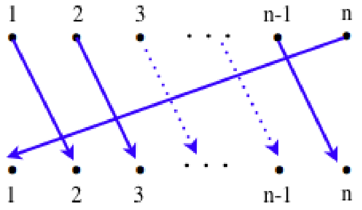
$$\varepsilon \leftrightarrow \left(\begin{array}{cccc} 1 & 2 & \dots & n \\ 1 & 2 & \dots & n \end{array} \right).$$

It may not seem obvious why we would want to consider the "do nothing" permutation, but we will consider this permutation quite a bit. As an analogy, think about 0, this is a symbol which represents "nothing" but yet appears almost everywhere in mathematics.

(b) An ***n*-cycle** is a permutation which cyclically permutes the values. For example,

$$\left(\begin{array}{cccccc} 1 & 2 & 3 & \dots & n-1 & n \\ 2 & 3 & 4 & \dots & n & 1 \end{array} \right).$$

We could also visualize this with an arrow diagram:



Every number moves to the right and the last one, n, cycles around back to 1.

3.3 Composing Permutations

We now look at how to combine two permutations in order to produce a third one. The method we use is called *composition*, or we'll sometimes refer to it as multiplication. This will be precisely the tool we will need in order to understand how two puzzle moves combine together to give a third.

Let α and β be two permutations of $[n]$. Recall that this means $\alpha, \beta : [n] \to [n]$ are both injective and surjective. We wish to define a new function $\alpha \circ \beta : [n] \to [n]$, called the *permutation composition*. In order to define a function on $[n]$ we just need to specify how it maps the elements. For $k \in [n]$ we'll define $(\alpha \circ \beta)(k)$ to be the result of first applying α, then applying β to the result. In other words,

$$(\alpha \circ \beta)(k) = \beta(\alpha(k)), \text{ for } k \in [n].$$

This new function is again a permutation. To see why we just need to observe that it is a bijection.

Injective: Suppose $(\alpha \circ \beta)(k) = (\alpha \circ \beta)(\ell)$ for some $k, \ell \in [n]$, then $\beta(\alpha(k)) = \beta(\alpha(\ell))$ implies $\alpha(k) = \alpha(\ell)$, since β is one-to-one. It follows that $k = \ell$ since α is one-to-one. Therefore, $\alpha \circ \beta$ is one-to-one.

Surjective: Consider any $m \in [n]$. Let $\ell \in [n]$ such that $\beta(\ell) = m$, and let $k \in [n]$ such that $\alpha(k) = \ell$. Both ℓ and k exist since α and β are onto. It follows that $(\alpha \circ \beta)(k) = \beta(\alpha(k)) = m$. Therefore, $\alpha \circ \beta$ is onto. This verifies that $\alpha \circ \beta$ is a permutation.

This way of combining permutations will essentially underline everything we do in this course so we should make this a formal definition. We will also drop the symbol $\circ$ to simplify writing.

Definition 3.4: Let $\alpha, \beta : [n] \to [n]$ be two permutations. The **permutation composition**, or **product**, of α and β is denoted by $\alpha\beta : [n] \to [n]$ is the permutation defined by:

$$\begin{array}{rccccc} \alpha\beta : & [n] & \to & [n] & \to & [n] \\ & k & \longmapsto & \alpha(k) & \longmapsto & \beta(\alpha(k)) \end{array}$$

The identity permutation ε, defined in Example 3.6a has the property that $\varepsilon\alpha = \alpha\varepsilon = \alpha$ for any permutation α.

Important: Notice that the composition is opposite to the way functions were combined in calculus. In calculus, and in most branches of mathematics, there is a long standing tradition that variables are to appear to the right of the function: $f(x)$. The composition, $(f \circ g)(x)$ is then read from right-to-left: $f(g(x))$. So why are we defining the composition of permutations as *left-to-right*, and going against long standing mathematical tradition? Imagine you were asked to apply the move sequence RF^{-1} to a Rubik's Cube. What move would you do first, R or F^{-1}? Popular convention is to read from left-to-right and apply R first, then F^{-1}. For example, this is how you are reading the words on the page right now, from left-to-right. This is precisely the convention we are using to combine permutations, we combine them from left-to-right.

Example 3.7: (a) Let $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}$ and $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix}$. Then

$$\alpha\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 2 & 5 & 1 & 3 \end{pmatrix}.$$

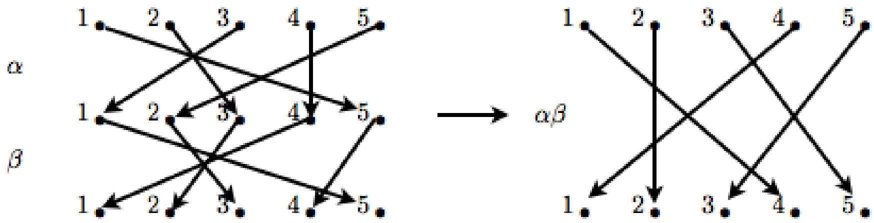
On the right we have 4 under 1, since $\alpha\beta(1) = \beta(\alpha(1)) = \beta(5) = 4$, so $\alpha\beta$ sends 1 to 4. This is illustrated by following the arrows above. Notice the movement is from left-to-right, which is our chosen convention for composing two permutation. The other values are determined in a similar fashion.

We can use Sage to multiply permutations.

Sage

```
sage: a=Permutation([5,3,1,4,2]);  a
[5, 3, 1, 4, 2]
sage: b=Permutation([5,3,2,1,4]); b
[5,3,2,1,4]
sage: a*b
[4, 2, 5, 1, 3]
```

We can also use the arrow diagram representation for permutations to give us more visual insight into how permutations are composed:



If we compose α and β in the other order, we find

$$\beta\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 3 & 5 & 4 \end{pmatrix}.$$

This shows that permutation composition is not commutative in general. That is, we typically have $\alpha\beta \neq \beta\alpha$.

Sage

```
sage:  b*a
[2, 1, 3, 5, 4]
sage:  a*b==b*a
False
```

(b) Let $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix}$ and $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 2 & 1 & 6 & 7 & 8 & 3 & 4 \end{pmatrix}$. Then

$$\alpha\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 2 & 1 & 6 & 7 & 8 & 3 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix} = \varepsilon.$$

Therefore $\alpha\beta$ is the identity permutation. Permutations with the property that their product is ε are called **inverse permutations**, since one permutation is undoing the rearrangement the other one performed.

(c) For any permutation α we can take the product of α with itself: $\alpha\alpha$, we write this as α^2. In general we write the product of α with itself n-times, $\alpha\alpha\cdots\alpha$, as α^n.

Suppose $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 1 & 5 & 2 \end{pmatrix}$, then the powers of α are:

$$\alpha^2 = \alpha\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 5 & 3 & 2 & 4 \end{pmatrix}, \qquad \alpha^3 = \alpha\alpha^2 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 2 & 1 & 4 & 5 \end{pmatrix}$$

$$\alpha^4 = \alpha\alpha^3 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 4 & 3 & 5 & 2 \end{pmatrix}, \qquad \alpha^5 = \alpha\alpha^4 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 5 & 1 & 2 & 4 \end{pmatrix}$$

$$\alpha^6 = \alpha\alpha^5 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}.$$

Check these products yourself. We see that α^6 is the identity permutation. This raises the question: Can we always multiply a permutation to itself a finite number of times and end up with the identity permutation?

From the previous example two questions are raised:

(i) For any permutation α, must there exist a permutation β such that $\alpha\beta = \varepsilon$?

(ii) For any permutation α, must there exist a positive integer n such that $\alpha^n = \varepsilon$?

If we think about a permutation as a move on one of our puzzles, say Rubik's Cube, then these questions are equivalent to asking: (i) When a move is applied, can it then be undone by another move? (ii) Applying the same move over and over again, will you eventually get back to where you started?

Phrased in this way, it may seem obvious that the answer is *yes* in both cases. For example, if the move F was applied (*clockwise* quarter turn of the front face), then the move F^{-1} undoes it (*counterclockwise* turn of the front face). Try this on your Rubik's Cube. Moreover, for the move F, applying it 4 times in a row takes you back to where you started. This means F^4 is the identity, or *do-nothing move*. If the answer to the questions above is now obvious then you already have a working understanding of *inverses* and *orders*.

We'll discuss these topics in a little more detail over the next few sections. But first let's play with the cube a little more.

Exercise 3.1: Consider Rubik's Cube and the legal moves F, B, R, L, U, D, F^{-1}, B^{-1}, R^{-1}, L^{-1}, U^{-1}, D^{-1}, and all successive combinations of these moves.

Recall a move sequence is read as follows: $F^{-1}U^2$ means first twist the front face a quarter turn in the counterclockwise direction, then turn the up face a half turn in the clockwise direction.

(a) What is the inverse of the move sequence $F^{-1}U^2$? That is, if you apply move sequence $F^{-1}U^2$, then what is the sequence of moves which will undo this?

(b) How many times does the move sequence U^2R^2 need to be applied in order to get you back to where you started? (Play with your cube to figure this out, and try not to lose count as you're twisting faces.)

Answer on page 37

3.4 Associativity of Permutation Composition

When adding and multiplying real numbers we don't need to worry about what to do first. For example, in the expression $2 \cdot 3 \cdot 4$ we get the same result if we multiply 2 and 3 first, then multiply the result by 4: $(2 \cdot 3) \cdot 4 = 6 \cdot 4 = 24$, as we get if we multiply 3 and 4 first, then multiply by 2: $2 \cdot (3 \cdot 4) = 2 \cdot 12 = 24$. This property of multiplication is called *associativity*, and it is written: $(ab)c = a(bc)$ for all $a, b, c \in \mathbb{R}$.

What associativity means is that we can write the product of three (or more) numbers without having to use grouping brackets: abc. Since no matter which product you take first it will not affect the result.

The same is true for addition of real numbers: $(a + b) + c = a + (b + c)$. This means we can write $a + b + c$ without any confusion about which sum to perform first.

A fundamental question to ask is: Is permutation composition associative? That is, must we have $(\alpha\beta)\gamma = \alpha(\beta\gamma)$?

The answer is yes, permutation composition *is* associative. Lucky for us, this means we don't have to use group brackets when writing long chains of products. The reason it is associative is simply because permutations are functions, and function composition is associative. To see why, consider permutations $\alpha, \beta, \gamma : [n] \to [n]$. For any $k \in [n]$,

$$((\alpha\beta)\gamma)(k) = \gamma((\alpha\beta(k)) = \gamma(\beta(\alpha(k))$$

and

$$(\alpha(\beta\gamma))(k) = (\beta\gamma)(\alpha(k)) = \gamma(\beta(\alpha(k)),$$

which are the same. [1] So $(\alpha\beta)\gamma = \alpha(\beta\gamma)$.

This means we can write $\alpha\beta\gamma$ for the product of these three permutations and there is no confusion about what product we should do first. The result won't change.

Example 3.8: Let $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}$, $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix}$, and $\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 4 & 3 & 5 \end{pmatrix}$. Then

$$\begin{aligned}(\alpha\beta)\gamma &= \left[\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix}\right]\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 4 & 3 & 5 \end{pmatrix} \\ &= \left[\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 2 & 5 & 1 & 3 \end{pmatrix}\right]\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 4 & 3 & 5 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 5 & 2 & 4 \end{pmatrix}\end{aligned}$$

and

$$\begin{aligned}\alpha(\beta\gamma) &= \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}\left[\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 2 & 1 & 4 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 1 & 4 & 3 & 5 \end{pmatrix}\right] \\ &= \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 1 & 4 & 2 \end{pmatrix}\left[\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 4 & 1 & 2 & 3 \end{pmatrix}\right] \\ &= \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 5 & 2 & 4 \end{pmatrix}\end{aligned}$$

It shouldn't come as a surprise that we get the same result for $(\alpha\beta)\gamma$ and $\alpha(\beta\gamma)$. This is what associativity means. We write this product as $\alpha\beta\gamma$.

3.5 Inverses of Permutations

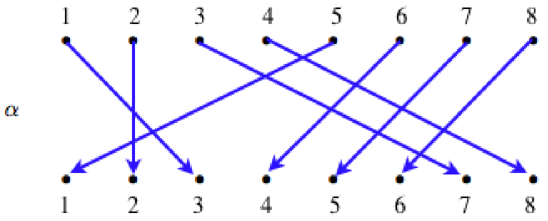
In Example 3.7(b) we saw two permutations α and β such that $\alpha\beta = \varepsilon$. We will call permutations with the property that their product is the identity, *inverses*. Let's look at this example a little more closely.

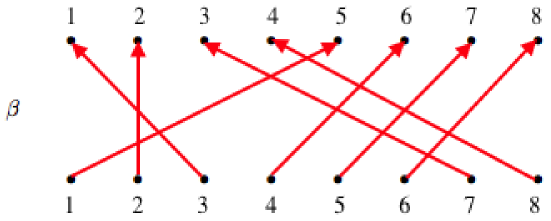
The permutations under consideration are:

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix} \text{ and } \beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 2 & 1 & 6 & 7 & 8 & 3 & 4 \end{pmatrix}.$$

We can represent α by an arrow diagram. Each blue arrow represents the mapping defined by the permutation α. If we replace each blue arrow with a red arrow pointing in the opposite direction then we get an arrow diagram representing β (follow arrows from bottom row to top row). In this sense, the inverse permutation is obtained by "reversing the arrows".

[1] Our convention is to compose permutations from left to right, see Definition 3.4.





We can do the same experiment with the array form of β. Let's flip β over, that is, we'll switch the top and bottom rows:

$$\begin{pmatrix} 5 & 2 & 1 & 6 & 7 & 8 & 3 & 4 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix},$$

then let's put the top row in increasing order, while keeping all the columns in tact:

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix}.$$

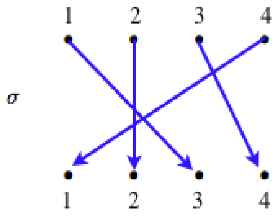
This precisely α! Should we be surprised this happened? What is really going on here?

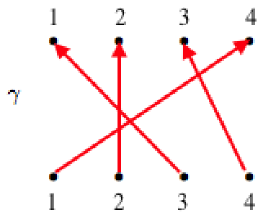
To see what is going on, let's recall that the notation means the number in the top row maps to the number directly beneath it in the bottom row. For instance, α maps 1 to 3. If β is to be the inverse of α then it must undo what α does. In particular, it must map 3 back to 1. This means 3 must appear above 1 in the array from of β. Let's say this again: if 1 is above 3 in α, then 3 is above 1 in β.

The same is true for every number. In general, we have if k is above m in α (i.e. $\alpha(k) = m$) then m is above k in β (i.e. $\beta(m) = k$). This explains exactly what we observed when we flipped β.

Now suppose, we start with a permutation, say $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 4 & 1 \end{pmatrix}$ and we flip the rows, and reorder the first row so it is increasing order, while keeping the columns in tact: $\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 2 & 1 & 3 \end{pmatrix}$. Is this a permutation? Well, each number from 1 to 4 appears in the second row, so it is surjective, and no number appears more than once, so it is injective. Therefore, yes, it is a permutation. And by the observation above, it is the inverse of σ, that is, $\sigma\gamma = \varepsilon$.

We can also use the arrow diagram to see this visually. γ was constructed by "reversing the arrows" of σ, so clearly γ is a bijection, and it is the inverse of σ, since it just undoes what σ is doing.





These observations tell us two things: every permutation has an inverse, and the inverse is unique. Moreover, we have a straightforward way to construct an inverse to a permutation given in array or arrow form.

This result is so important that we state it as a theorem. We'll also give a formal proof of the theorem, which captures the essence of our discussion above in just a few lines.

Theorem 3.1: *For any permutation $\alpha : [n] \to [n]$, there exists a unique permutation $\beta : [n] \to [n]$ such that $\alpha\beta = \beta\alpha = \varepsilon$.*

Proof: Let α be a permutation, define a new function $\beta : [n] \to [n]$ as follows:

$$\beta(m) = k \iff \alpha(k) = m$$

for $k, m \in [n]$. Since α is bijective, for any m such a k exists and is unique so β is well defined. It follows that $(\alpha\beta)(k) = \beta(\alpha(k)) = \beta(m) = k$ and $(\beta\alpha)(m) = \alpha(\beta(m)) = \alpha(k) = m$. This proves the theorem. □

Definition 3.5: For any permutation α the unique permutation β such that $\alpha\beta = \beta\alpha = \varepsilon$ is called the **inverse** of α and is denoted by α^{-1}.

Example 3.9: Find the inverse of each of the following permutations. Verify it is the inverse by computing the product and showing it is the identity permutation.

(a) $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 2 & 5 & 4 \end{pmatrix}$

(b) $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$

(a) The inverse of α can be obtained by reading the array form from the bottom row to the top row. For example, 1 in the bottom row must map to the number above it, which is 2. Similarly for the other numbers, so $\alpha^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 1 & 5 & 4 \end{pmatrix}$.

Sage has a built-in `inverese()` command.

Sage

```
sage:  a=Permutation([3,1,2,5,4])
sage:  a.inverse()
[2, 3, 1, 5, 4]
```

(b) Similar to (a), we read the array form of β from bottom-to-top to get the array form of β^{-1}: $\beta^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$. Notice this is just β itself. Therefore β is its own inverse.

Sage

```
sage:  b=Permutation([3,4,1,2])
sage:  b.inverse()
[3, 4, 1, 2]
```

3.5.1 Inverse of a Product

Apply the move sequence RU to your Rubik's Cube. Now undo this move sequence. That is, return the cube to the state it was in before you apply RU. It is very likely you just applied the move sequence $U^{-1}R^{-1}$. If you did, then you have a working understanding of how to find the inverse of a product.

As another example, in the morning you get dressed you put on your *socks* then your *shoes*, but when you come home at night and get undressed you takes off your *shoes* then your *socks*. The order in which things are undone is opposite to which they were done.

If these two example seem obvious, it is because they in fact are. But even obvious things can be stated as theorems, which are just convenient summaries of observations for later use.

Theorem 3.2: *For two permutations α and β,*

$$(\alpha\beta)^{-1} = \beta^{-1}\alpha^{-1}.$$

In general, the inverse of a product of permutations is the product of the inverses in the reverse order:

$$(\alpha_1\alpha_2\cdots\alpha_k)^{-1} = \alpha_k^{-1}\cdots\alpha_2^{-1}\alpha_1^{-1}.$$

Proof: Taking the product, and using associativity of permutation multiplication,

$$\begin{aligned}(\alpha\beta)(\beta^{-1}\alpha^{-1}) &= \alpha\beta\beta^{-1}\alpha^{-1}\\ &= \alpha\varepsilon\alpha^{-1}\\ &= \alpha\alpha^{-1}\\ &= \varepsilon\end{aligned}$$

Therefore, $\beta^{-1}\alpha^{-1}$ is the inverse of $\alpha\beta$. A similar argument proves the general statement. □

3.5.2 Cancellation Property

An important property of the real numbers that we use all the time is the ability to cancel the same (non-zero) factor on both sides of an equation. For example if $2x = 6$ then $2x = 2 \cdot 3$ and we cancel the 2's to get $x = 3$. The reason we could "cancel" the $2's$ is simply because we could multiply both sides of the equation by the inverse of 2, namely $1/2$. That is $(1/2)(2x) = (1/2)(2 \cdot 3)$, which means $[(1/2)2]x = [(1/2)2]3$ (note the use of associativity of multiplication here), and so $x = 3$.

Luckily, this familiar property also holds for permutations.

Lemma 3.3 (Cancellation Property): *If $\alpha, \beta, \gamma \in S_n$ where $\alpha\beta = \alpha\gamma$ then $\beta = \gamma$. Similarly, if $\beta\alpha = \gamma\alpha$ then $\beta = \gamma$.*

Proof: Multiplying both sides of $\alpha\beta = \alpha\gamma$ on the left by α^{-1} we get

$$\alpha^{-1}(\alpha\beta) = \alpha^{-1}(\alpha\gamma).$$

By associativity

$$(\alpha^{-1}\alpha)\beta = (\alpha^{-1}\alpha)\gamma.$$

and so

$$\varepsilon\beta = \varepsilon\gamma,$$

which means $\beta = \gamma$.

A similar argument shows the right cancellation property as well. □

As a consequence of the cancellation property the identity permutation is the *only* permutation that when multiplied by another permutation leaves it unchanged. That is, it has the property that $\alpha\varepsilon = \alpha$ for any $\alpha \in S_n$. To see this, suppose β is a permutation with this property too, that is $\alpha\beta = \alpha$ for some α. Then $\alpha\beta = \alpha\varepsilon$, and by cancellation of α we have $\beta = \varepsilon$.

3.6 The Symmetric Group S_n

The set of all permutations of the set $[n]$ is called the *symmetric group of degree n*, and is denoted by S_n. In other words,

$$S_n = \{\alpha \mid \alpha \text{ is a permutation of } [n]\}.$$

We've already seen that elements of S_n can be written in the form

$$\begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \alpha(1) & \alpha(2) & \alpha(3) & \dots & \alpha(n) \end{pmatrix}.$$

It is straightforward to compute the cardinality of the set S_n. There are n choices for $\alpha(1)$. Once $\alpha(1)$ has been chosen, there are $n-1$ possibilities for $\alpha(2)$ (since α is injective we must have $\alpha(1) \neq \alpha(2)$). Once $\alpha(2)$ has been chosen there are $n-2$ choices for $\alpha(3)$. Continuing in this way we see that there are $n \cdot (n-1) \cdot (n-2) \cdots 3 \cdot 2 \cdot 1 = n!$ possible choices for $\alpha(1)$ to $\alpha(n)$. Each choice gives a different permutation. Therefore $|S_n| = n!$.

Let's summarize what we know so far about S_n.

- S_n, the symmetric group of degree n, is the set of all permutation of $[n] = \{1, 2, \dots, n\}$.
- $|S_n| = n!$
- Two elements $\alpha, \beta \in S_n$ can be composed (multiplied) to give another element $\alpha\beta \in S_n$.[2]
- The *identity* permutation is $\varepsilon = \begin{pmatrix} 1 & 2 & ... & n \\ 1 & 2 & ... & n \end{pmatrix}$. It has the property that $\varepsilon\alpha = \varepsilon\alpha = \alpha$ for all $\alpha \in S_n$.
- Every $\alpha \in S_n$ has an inverse denoted by α^{-1}. The defining property of an inverse is $\alpha\alpha^{-1} = \alpha^{-1}\alpha = \varepsilon$.
- $(\alpha_1\alpha_2\cdots\alpha_k)^{-1} = \alpha_k^{-1}\cdots\alpha_2^{-1}\alpha_1^{-1}$.
- Permutation composition (multiplication) is associative: $\alpha(\beta\gamma) = (\alpha\beta)\gamma$.
- Permutation composition (multiplication) is not necessarily commutative.
- Cancellation Property: $\alpha\beta = \alpha\gamma$ implies $\beta = \gamma$, and $\beta\alpha = \gamma\alpha$ implies $\beta = \gamma$.

3.7 Rules for Exponents

When we describe moves on Rubik's Cube we'll write things like: RB^2R^{-1}. Exponents are serving two purposes here: (i) they represent inverse moves, R^{-1} is the inverse of R, (ii) they represent repetition of moves, B^2 is the move B repeated twice.

If we follow the move sequence $RU^{-2}B^2R^{-1}DU^{-2}$ with the move U then the complete move sequence would be

$$RU^{-2}B^2R^{-1}DU^{-2}U.$$

But certainly, $U^{-2}U$ simplifies to U^{-1}, since a counterclockwise half turn U^{-1} followed by a clockwise quarter turn U is equivalent to a counterclockwise quarter turn. This means the complete move sequence is equivalent to

$$RU^{-1}B^2R^{-1}DU^{-1}.$$

[2] the convention of these notes is to compose permutations from left-to-right,

We write this as $(RU^{-2}B^2R^{-1}DU^{-2})U = RU^{-2}B^2R^{-1}DU^{-1}$.

This notation translates nicely to composition of permutations.

If $\alpha \in S_n$ and m is a positive integer then α^m denotes the product of α with itself m-times. That is, $\alpha^m = \alpha\alpha\cdots\alpha$.

We define negative exponents by the rule $\alpha^{-m} = (\alpha^{-1})^m$, where m is any positive integer.

We define the zero exponent by $\alpha^0 = \varepsilon$, where ε is the identity permutation.

An important observation is that some of the familiar "rules of exponents" apply to the composition of permutations. Specifically, for any two integers m and k and for any $\alpha \in S_n$, we have

(a) $\alpha^m\alpha^k = \alpha^{m+k}$

(b) $(\alpha^m)^k = \alpha^{mk}$

This follows precisely from the fact that we are defining an exponent to represent repeated composition.

One property that you may be familiar with from multiplication of real numbers is: $(ab)^m = a^m b^m$. This is *not* true for permutations: if $\alpha, \beta \in S_n$ and $m \in \mathbb{Z}$ then in general $(\alpha\beta)^m$ is not equal to $\alpha^m\beta^m$.

For real numbers this property relies on the fact that multiplication of real numbers is *commutative*. We've already seen this is not the case for permutations under composition.

However, we do have the following result.

Lemma 3.4: *If $\alpha, \beta \in S_n$ commute with each other, that is $\alpha\beta = \beta\alpha$, then for all integers m, $(\alpha\beta)^m = \alpha^m\beta^m$.*

Proof: When $m = 0$ or 1 the result is trivial. Let's now consider the case when $m \geq 2$. We will use mathematical induction on m to prove the result. Starting with the base case $m = 2$ we see that

$$\begin{aligned}(\alpha\beta)^2 = (\alpha\beta)(\alpha\beta) &= \alpha(\beta\alpha)\beta \quad \text{by associativity,}\\ &= \alpha(\alpha\beta)\beta \quad \text{since } \alpha \text{ and } \beta \text{ commute,}\\ &= \alpha^2\beta^2.\end{aligned}$$

Therefore the statement of the lemma holds for $m = 2$. For the inductive hypothesis we assume the result holds for $m = k$ and prove it must then hold for $m = k + 1$.

$$\begin{aligned}(\alpha\beta)^{k+1} &= (\alpha\beta)(\alpha\beta)^k\\ &= (\alpha\beta)(\alpha^k\beta^k) \quad \text{by the induction hypothesis,}\\ &= \alpha(\beta\alpha^k)\beta^k \quad \text{by associativity,}\\ &= \alpha(\alpha^k\beta)\beta^k \quad \text{since } \alpha \text{ and } \beta \text{ commute,}\\ &= \alpha^{k+1}\beta^{k+1} \quad \text{by associativity.}\end{aligned}$$

Therefore, by mathematical induction, the result follows for all $m \geq 2$. Combined with our initial observations we see that the statement of the lemma holds for all $m \geq 0$.

For the case when m is negative, first notice that if α and β commute then so do α^{-1} and β^{-1}. To see this just take inverses of each side of the equation $\alpha\beta = \beta\alpha$. Now if $m = -k$ where k is a positive integer then

$$\begin{aligned}
(\alpha\beta)^m &= ((\alpha\beta)^{-1})^k \quad \text{by definition of a negative exponent,}\\
&= (\beta^{-1}\alpha^{-1})^k\\
&= (\alpha^{-1}\beta^{-1})^k \quad \text{since } \alpha^{-1} \text{ and } \beta^{-1} \text{ commute}\\
&= (\alpha^{-1})^k(\beta^{-1})^k \quad \text{by the lemma applied to } \alpha^{-1}, \beta^{-1} \text{ and } k > 0,\\
&= \alpha^{-k}\beta^{-k} = \alpha^m\beta^m.
\end{aligned}$$

This proves the lemma.

□

3.8 Order of a Permutation

The **order** of a permutation $\alpha \in S_n$ is the smallest positive integer m such that $\alpha^m = \varepsilon$.

In Example 3.7 we saw that for $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 1 & 5 & 2 \end{pmatrix}$ the smallest m for which $\alpha^m = \varepsilon$ is 6. We say α has *order* 6, and we write $\mathrm{ord}(\alpha) = 6$.

As another example, $\beta = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$ is an element in S_3 of order 2, since $\beta \neq \varepsilon$, but $\beta^2 = \varepsilon$.

Must every permutation have a finite order? The next theorem answers this question.

Theorem 3.5: *For any $\alpha \in S_n$ there exists a positive number m for which $\alpha^m = \varepsilon$. The smallest such m is the* ***order*** *of α, denoted ord(α).*

Proof: Consider the set of all powers of α, $\{\alpha^k : k \in \mathbb{Z}^+\}$. Since this is a subset of the finite set S_n it must also be finite. This means all the powers of α cannot be distinct, so there must be k, ℓ such that $\alpha^k = \alpha^\ell$ where $k > \ell > 0$. Now multiplying $\alpha^{-\ell}$ to the left of both sides (i.e. cancelling α^ℓ) we get:

$$\alpha^{-\ell}\alpha^k = \alpha^{-\ell}\alpha^\ell$$

and so

$$\alpha^{k-\ell} = \varepsilon.$$

This proves the theorem. □

We can now describe precisely which integers m have the property that $\alpha^m = \varepsilon$.

Theorem 3.6: *Let α be a permutation. If $\alpha^m = \varepsilon$ then ord(α) divides m.*

Proof: Let $n = \mathrm{ord}(\alpha)$, and suppose $\alpha^m = \varepsilon$. By the division algorithm (Theorem B.1 in Appendix B) there exist integers q and $0 \leq r < m$ such that $m = qn + r$. In other words, n goes into m q-times, with r left over. Therefore

$$\varepsilon = \alpha^m = \alpha^{qn+r} = (\alpha^n)^q\alpha^r = \varepsilon^q\alpha^r = \alpha^r.$$

Since r is smaller than the order of α this is only possible if $r = 0$. Hence n divides m. □

Exercise 3.2: Let

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 2 & 1 & 4 & 3 \end{pmatrix}, \quad \beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 4 & 1 & 2 & 3 \end{pmatrix}, \quad \gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 1 & 3 & 5 \end{pmatrix}.$$

Determine the order of (a) α (b) β (c) β^{-1} (d) γ (e) $\alpha^{-1}\gamma\alpha$.

Answer on page 37

Exercise 3.3: Consider Rubik's Cube. Determine the orders of each of the following moves by physically doing the move successively on the cube. It is best to start with your cube in the solved state so you can easily recognize when you've returned to that state.

(a) R

(b) $R^2L^2U^2$

(c) U^2R

(d) UR

Answer on page 37

If you stuck with it long enough, and didn't lose count, you would find that UR has order 105. That means you would have to apply UR a total of 105 times (or a total of 210 quarter face turns) before you get back to where you started.

One of our goals will be to thoroughly understand orders of move sequences: specifically how to compute the order of a move sequence without having to physically manipulate the cube.

For example, the move sequence $RU^2D^{-1}BD^{-1}$ has order 1260. We'll soon see how to compute this rather quickly using Sage.

3.9 Exercises

1. Show that a function from a finite set A to itself is one-to-one if and only if it is onto. Is this true when A is infinite?
2. Suppose A and B are finite sets and $|A| > |B|$. Is there an *injective* function $f : A \to B$? Explain.
3. For $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 1 & 2 & 4 \end{pmatrix}$, $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 1 & 2 \end{pmatrix}$, and $\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}$ verify that $\alpha(\beta\gamma) = (\alpha\beta)\gamma$. This provides some experimental evidence for the associative law.
4. Consider the following permutations

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 4 & 6 & 7 & 1 & 5 & 8 & 2 \end{pmatrix}, \quad \beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{pmatrix},$$

$$\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 4 & 6 & 7 & 1 & 3 & 2 & 8 \end{pmatrix}.$$

Determine each of the following.

(a) $\alpha\beta$

(b) $\alpha\gamma\beta$

(c) β^{-1}

(d) $(\gamma\beta)^{-1}$

(e) $\beta^{-1}\gamma^{-1}$

(f) $\alpha^{-1}\gamma\alpha$

(g) ord(α)

(h) ord(β)

(i) ord$(\alpha^{-1}\gamma\alpha)$

5. Find the inverse of each of the following permutations. Verify the product you found actually is the inverse by computing the product and showing it is the identity permutation.

   (a) $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 2 & 1 \end{pmatrix}$ (b) $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 4 & 1 & 5 & 7 & 3 & 8 & 2 & 6 \end{pmatrix}$

6. For $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 3 & 4 \end{pmatrix}$ explain how you know $\alpha^{2011} \neq \varepsilon$, without actually computing all 2011 powers of α.

7. Show that an n-cycle $\begin{pmatrix} 1 & 2 & 3 & \dots & n \\ n & 1 & 2 & \dots & n-1 \end{pmatrix}$ has order n.

8. Show that for any $\alpha \in S_n$, ord$(\alpha) =$ ord(α^{-1}).

9. **There is always something that doesn't commute.** Show that if $n \geq 3$, then for every element α in S_n, if α is not the identity permutation ε then there is some other permutation β in S_n with which α does not commute: $\alpha\beta \neq \beta\alpha$.

10. For any permutations α and β and any integer n show that $(\alpha^{-1}\beta\alpha)^n = \alpha^{-1}\beta^n\alpha$.

11. For $\alpha, \beta \in S_n$ show that if $(\alpha\beta)^2 = \alpha^2\beta^2$ then α commutes with β: that is, $\alpha\beta = \beta\alpha$.

12. Show that if $\alpha\beta\gamma\beta^{-1}\alpha = \alpha\beta\sigma\beta^{-1}\alpha$ then $\gamma = \sigma$.

13. Show that the number of elements α in S_n such that $\alpha^3 = \varepsilon$ is odd. In other words, show the set $\{\alpha \in S_n \mid \alpha^3 = \varepsilon\}$ has odd cardinality.

Answers to in-chapter exercises:

Exercise 3.1: (a) $U^{-2}F$ which could also be written as U^2F since U^2 and U^{-2} are equivalent moves. (b) 6

Exercise 3.2: (a) 3 (b) 6 (c) 6 (d) 4 (e) 4

Exercise 3.3: (a) 4 (b) 4 (c) 30 (d) 105

Lecture 4

Permutations: Cycle Notation

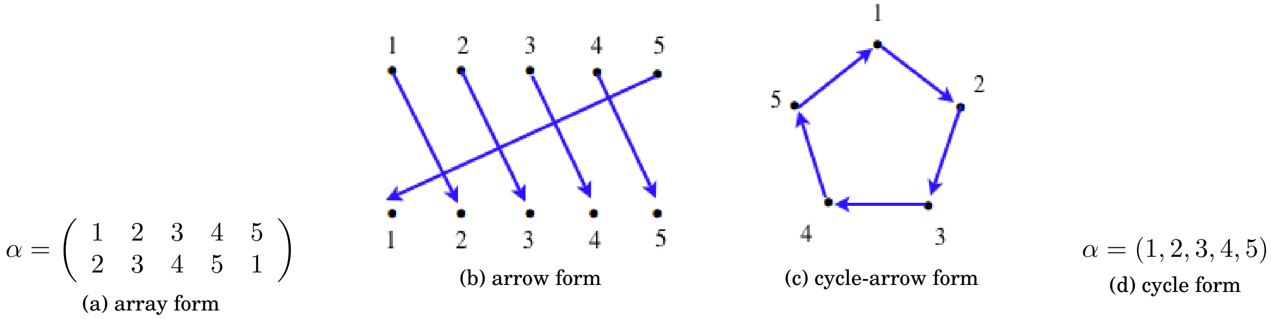
In this section we introduce a simple, yet extremely powerful, notation for permutations: *cycle form*. We'll revisit the concepts of products (composition), order, and inverses, and see how our new notation simplifies calculations.

4.1 Permutations: Cycle Notation

Consider the 5-cycle permutation α defined as follows:

$$\alpha(1) = 2,\ \ \alpha(2) = 3,\ \ \alpha(3) = 4,\ \ \alpha(4) = 5,\ \ \alpha(5) = 1.$$

The *array form* of α is shown in Figure 4.1a, and the *arrow diagram* is shown in Figure 4.1b.



$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$$

(a) array form

(b) arrow form

(c) cycle-arrow form

$$\alpha = (1, 2, 3, 4, 5)$$

(d) cycle form

Figure 4.1: Different representations for a 5-cycle.

Another diagram which provides a visual representation of the structure of the permutation is shown in Figure 4.1c, it is called the *cycle-arrow form*. In this diagram all the information for α is still present. For example, to determine $\alpha(3)$ look at the diagram and find 3, then see where the arrow takes it. In this case it takes 3 to 4, so $\alpha(3) = 4$.

There are a few nice things about cycle-arrow form: (1) it displays visually the cycle structure (i.e. we can see the 5 numbers cycling around in a circular fashion, which is why we called it a 5-cycle), and (2) it uses only one set of numbered dots, making the diagram more compact than our original arrow form.

Though mathematically satisfactory, the cycle-arrow form is cumbersome to draw. However, leaving out the arrows we can simply write the 5-cycle as:

$$\alpha = (1, 2, 3, 4, 5)$$

This represents that fact that α maps each number to the next one in the list, and maps 5 back around to the start of the list, which is 1. This representation is shown in Figure 4.1d and it is called *cycle form*.

All representations in Figure 4.1 have their own benefits, but it is this cycle form that is the most compact, and this will be the form we primarily use in this book.

When working with cycle form, $\alpha = (1, 2, 3, 4, 5)$, you should read it as follows:

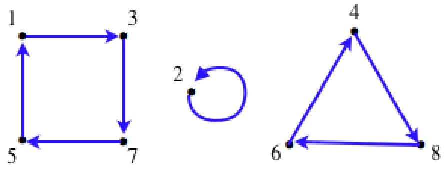
"1 goes to 2, 2 goes to 3, 3 goes to 4, 4 goes to 5, and 5 goes to 1."

We don't need to start at 1 when writing down the cycle form. If we started at 3, for instance, and constructed the list of numbers we visit by traveling around Figure 4.1c then we get $(3, 4, 5, 1, 2)$. This is another perfectly acceptable representation of α: reading this cycle notation as described above will tell us exactly how α acts as a function. In particular, we can represent α by any of the equivalent cycle forms:

$$\alpha = (1, 2, 3, 4, 5) = (2, 3, 4, 5, 1) = (3, 4, 5, 1, 2) = (4, 5, 1, 2, 3) = (5, 1, 2, 3, 4).$$

Despite this notation allowing for non-unique representations of permutations, there is an easy fix. Just write the cycle so that the first number is the smallest number in the cycle. In this case we would then write $\alpha = (1, 2, 3, 4, 5)$ since 1 is the smallest number in this cycle.

Let's look at another permutation: $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix}$. The cycle arrow form is:



This reveals so much about the permutation, especially when you imagine taking powers of it: β^n. For instance, 1,3,5,7 only get permuted amongst themselves, so there is no k such that $\beta^k(1) = 4$. Also, since a 4-cycle has order 4, then β^4 would leave 1,3,5,7 untouched: $\beta^4(x) = x$ when $x = 1, 3, 5, 7$. This means β^4 is a 3-cycle: $\beta^4 = (4, 8, 6)$.

To construct the cycle form of β we look at the arrow form above and notice that 1 goes to 3, 3 goes to 7, 7 goes to 5 and 5 goes back to 1. This can simply be written as $(1, 3, 7, 5)$. Similarly, 2 goes to 2 so we write this as (2), and the $4, 6, 8$ triangle can be written as $(4, 8, 6)$. Therefore, the cycle form of β is

$$\beta = (1, 3, 7, 5)(2)(4, 8, 6).$$

This is a compact way to represent the permutation β, and we haven't lost any information. For example, we can use the cycle form to determine $\beta(3)$ by noticing in $(1, 3, 7, 5)(2)(4, 8, 6)$ the number 3 is followed by 7, so $\beta(3) = 7$. Similarly, $\beta(5) = 1$ since from 5 we wrap around in the cycle and get back to 1.

If we make one further convention: *to leave off any number that gets mapped to itself*, then β can be written in an even more compact form:

$$\beta = (1, 3, 7, 5)(4, 8, 6).$$

With this convention, any number not present in the cycle form is assumed to map back to itself.

Recall a permutation of the form $(a_1, a_2, \ldots, a_m)$ is called an m- *cycle*. We would say β is the product of a 3-cycle and a 4-cycle.

Example 4.1: To determine the cycle form of the permutation

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 5 & 1 & 6 & 8 & 4 & 10 & 7 & 2 & 9 & 3 \end{pmatrix}$$

start with the smallest number in the set, in this case it is 1. Since $\alpha(1) = 5$ we begin the cycle by writing

$$(1, 5, \ldots) \ldots.$$

Next, 5 maps to 4, so we continue building the cycle

$$(1, 5, 4, \ldots) \ldots.$$

Continuing in this way we construct $(1, 5, 4, 8, 2, \ldots) \ldots$, and since 2 maps back to 1 then we close off the cycle:

$$(1, 5, 4, 8, 2) \ldots.$$

Next, we pick the smallest number that doesn't appear in any previously constructed cycle. This is the number 3 in this case. We now repeat what we just did and construct the cycle involving 3:

$$(1, 5, 4, 8, 2)(3, 6, 10) \ldots.$$

We now pick the smallest number that doesn't appear in any previously constructed cycle, which is 7, and construct the cycle to which it belongs. In this case 7 just maps to itself:

$$(1, 5, 4, 8, 2)(3, 6, 10)(7) \ldots.$$

Finally, the only number remaining is 9 and it maps back to itself so the cycle for of α is

$$(1, 5, 4, 8, 2)(3, 6, 10)(7)(9)$$

which simplifies to

$$\alpha = (1, 5, 4, 8, 2)(3, 6, 10)$$

since our convention is to omit 1-cycles. Therefore, α is the product of a 3-cycle and a 5-cycle.

Exercise 4.1: **Converting from array to cycle form.** Convert the permutation given in array form

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$$

to cycle form.

Answer on page 49

Exercise 4.2: **Converting from cycle to array form.** For the permutation given in cycle form by $(1, 3, 5, 2)(4, 7) \in S_8$, express it in array form.

Answer on page 49

4.2 Products of Permutations: Revisited

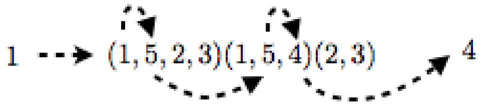
It is not efficient to convert permutations from cycle form to array from, then compose the permutations in array form, only to convert back to cycle form. Instead, we will work entirely with the cycle form but we do so by *thinking* of their representation in array form.

For example, consider the permutations $\alpha = (1,5,2,3)$ and $\beta = (1,5,4)(2,3)$ in S_5. What is the cycle form of $\alpha\beta$? Of course, we could just stick the two permutations together, end-to-end, and write

$$\alpha\beta = (1,5,2,3)(1,5,4)(2,3)$$

but it will be more convenient to express the permutation in *disjoint cycle form*, that is where the various cycles have no numbers in common.

We determine the cycle form of $\alpha\beta$ by determining exactly how it maps each number, beginning with 1. Keep in mind that permutation composition is done from left-to-right, and each cycle that does not contain a number fixes that number. We have that $(1,5,2,3)$ sends 1 to 5, $(1,5,4)$ sends 5 to 4, and $(2,3)$ fixes 4. So the effect of $\alpha\beta$ is it sends 1 to 4.



Thus we begin writing the disjoint cycle form as $\alpha\beta = (1,4,\ldots)\ldots$.

Repeating this process with 4, we have, cycle-by-cycle, left-to-right,

$$4 \xrightarrow{(1,5,2,3)} 4 \xrightarrow{(1,5,4)} 1 \xrightarrow{(2,3)} 1,$$

so that $\alpha\beta(4) = 1$, and the cycle form is now $\alpha\beta = (1,4)\ldots$.

Next we pick the smallest number that is not in any previously constructed cycle, this would be 2. Repeating this process with 2, cycle-by-cycle, left-to-right,

$$2 \xrightarrow{(1,5,2,3)} 3 \xrightarrow{(1,5,4)} 3 \xrightarrow{(2,3)} 2,$$

so that $\alpha\beta(2) = 2$, and the cycle for is now $\alpha\beta = (1,4)(2)\ldots$.

Continuing in this way we find that $\alpha\beta = (1,4)(2)(3,5) = (1,4)(3,5)$.

The important thing to keep in mind when multiplying cycles is to *keep moving* from one cycle to the next from left-to-right.

Example 4.2: Let $\alpha = (1,4,6,3,7)(2,8)$ and $\beta = (2,5,3)(4,7,8,1)$ be permutations in S_8. Then

$$\alpha\beta = (1,4,6,3,7)(2,8)(2,5,3)(4,7,8,1) = (1,7,4,6,2)(3,8,5)$$

and

$$\beta\alpha = (2,5,3)(4,7,8,1)(1,4,6,3,7)(2,8) = (1,6,3,8,4)(2,5,7).$$

Check this yourself. To start off, consider what happens to 1 under $\alpha\beta$:

$$1 \xrightarrow{(1,4,6,3,7)} 4 \xrightarrow{(2,8)} 4 \xrightarrow{(2,5,3)} 4 \xrightarrow{(4,7,8,1)} 7,$$

so $(\alpha\beta)(1) = 7$.

4.3 Properties of Cycle Form

Two basic properties of permutations are: (a) **every permutation can be written as a product of disjoint cycles**, and (b) **disjoint cycles commute**.

The first property was implicit in our discussion of how to construct the cycle form of a permutation. In particular, when we finished constructing a cycle, the first thing we did was look for a number that did not appear in a previously constructed cycle. This guarantees that our cycles will be disjoint.

The second property: *disjoint cycles commute*, is also fairly straightforward consequence of the disjoint cycle notation. For example, consider the disjoint cycles $\alpha = (1,3,2)$ and $\beta = (4,5)$. When multiplying these cycles it doesn't matter which order the product is taken: $\alpha\beta = (1,3,2)(4,5) = (4,5)(1,3,2) = \beta\alpha$. Both of these products represent the same permutation: $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 2 & 5 & 4 \end{pmatrix}$. As a former student of mine once said, *it is kind of like two games of musical chairs going on in two different rooms, neither one has any influence on the other.*

Even though this property is straightforward, it is very important, so we will state it as a theorem.

Theorem 4.1 (Disjoint Permutations Commute): *If $\alpha, \beta \in S_n$ and have no numbers in $[n]$ that are moved by both α and β then $\alpha\beta = \beta\alpha$. In other words, if the disjoint cycle form of α has no number in common with the disjoint cycle form of β then α and β commute.*

For a more physical example of disjoint cycles commuting consider the moves R and L of Rubik's cube. These moves are disjoint in the sense that their is no common piece that is moved by both R and L. Notice that RL and LR result in exactly the same position of the cube, so in this sense RL = LR, and so R and L commute.

4.4 Order of a Permutation: Revisited

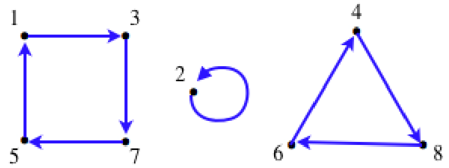
Recall the **order** of a permutation $\alpha \in S_n$ is the smallest positive integer m such that $\alpha^m = \varepsilon$. (See Section 3.8.) To determine the order of a given permutation our only technique so far was to just continue computing powers until we hit the identity. This is a very inefficient way to compute the order.

The disjoint cycle form has the enormous advantage of allowing us to "eyeball" the order of a permutation.

For example the 5-cycle $(1,2,3,4,5)$ has order 5. In general, an m-cycle has order m. (You are asked to show this in Exercise 9, and were also asked this in Lecture 3 Exercise 7.) The order of a product of disjoint cycles is given by the next theorem.

Theorem 4.2 (Order of a Permutation): *The order of a permutation written in disjoint cycle form is the least common multiple of the lengths of the cycles.*

Before we prove this theorem let's see why it should be true. Consider the permutation $\beta = (1,3,7,5)(4,8,6)$, which is the product of a cycle of length 3 and a cycle of length 4. The arrow diagram is as follows.



We want to determine the smallest power k so that β^k is the identity. Every application of β moves the numbers around the square (4-cycle) one position, so in order to have numbers return to their original position β must be applied 4, or a multiple of 4, times. This means $4 \mid k$. [1] Similarly, considering the triangle (3-cycle) β would need to be applied a multiple of 3 times to move numbers back to their original positions. This means $3 \mid k$. Since we require both 3 and 4 to divide k, and we want k to be as small as possible, this means k is the *least common multiple* of 3 and 4, that is $\text{ord}(\beta) = k = \text{lcm}(3,4) = 12$. Sure enough, if we check we find $\beta^{12} = \varepsilon$.

An algebraic way to see $\beta^{12} = \varepsilon$ is as follows:

$$\beta^{12} = [(1,3,7,5)(4,8,6)]^{12} = (1,3,7,5)^{12}(4,8,6)^{12} = [(1,3,7,5)^4]^3[(4,8,6)^3]^4 = \varepsilon^3\varepsilon^4 = \varepsilon.$$

Here we used the fact that an m-cycle has order m, and $(\sigma_1\sigma_2)^k = \sigma_1^k\sigma_2^k$, for *disjoint* cycles σ_1 and σ_2 (recall that disjoint cycles commute by Theorem 4.1).

This is precisely the idea that we use to give a general proof of the theorem.

Proof: (Theorem 4.2)

One cycle: As we noted above, a cycle of length m has order m. (See Exercise 9.)

Two disjoint cycles: Now suppose α and β are disjoint cycles of lengths a and b. Let k be the least common multiple of a and b, that is, k is the smallest positive integer which is divisible by both a and b. Since α and β commute then $(\alpha\beta)^k = \alpha^k\beta^k = \varepsilon$ (here we used Lemma 3.4 and that fact that $a|k$ implies $\alpha^k = \varepsilon$ and $b|k$ implies $\beta^k = \varepsilon$). It follows from Theorem 3.6 that the order of $\alpha\beta$, call it t, divides k. We now wish to show $t = k$. From $\varepsilon = (\alpha\beta)^t = \alpha^t\beta^t$ it follows that $\alpha^{-t} = \beta^t$. However, α and β have no symbol in common, and since raising a cycle to a power does not introduce new symbols, α^{-t} and β^t also have no symbol in common. Since $\alpha^{-t} = \beta^t$ and have no common symbols then they both must be the identity: $\alpha^{-t} = \beta^t = \varepsilon$. If follows from Theorem 3.6 that t is divisible by a and b. This means that $k = \text{lcm}(a,b)$ must also divide t. Therefore $t = k$, as desired.

More than two disjoint cycle: The general case involving more than two cycles is handled in an analogous way. □

Example 4.3: (a) The order of $\alpha = (1,3,4)(2,5)$ is $\text{lcm}(3,2) = 6$. Observe that

$$\alpha^6 = [(1,3,4)(2,5)]^6 = (1,3,4)^6(2,5)^6 = \varepsilon.$$

(b) The permutation $\beta = (1,7,4,10,3)(2,5,6,9)(8,11)$ has order $\text{lcm}(5,4,2) = 20$. Notice how quickly we were able to compute this order. If we tried to do it by successively computing powers of β we would need to compute 20 powers, and this assumes we didn't make any mistakes in the tedious calculations. This shows the power of Theorem 4.2.

Exercise 4.3: Find the order of each of the following permutations:
(a) $(1,3)$ **(b)** $(1,5,2,3)$ **(c)** $(1,5,3,7)(2,6,8)$

Answer on page 49

4.5 Inverse of a Permutation: Revisited

Every permutation can be written as a product of disjoint cycles: $\alpha = \sigma_1\sigma_2\cdots\sigma_k$. We have already seen that the inverse of a product is the product of the inverses in the reverse order, so

$$\alpha^{-1} = \sigma_k^{-1}\cdots\sigma_2^{-1}\sigma_1^{-1}.$$

This means, in order to determine α^{-1} directly from its cycle form we just need to know how to find the inverse of a cycle.

[1] For integers, the vertical bar $\mid$ means "divides", so $a \mid b$ is read "a divides b" and means $b = ak$ for some integer k. (Appendix B)

Consider the 5-cycle $\alpha = (1,2,3,4,5)$. We'd like to come up with a simple method for determining the inverse α^{-1} directly from the cycle form, and without having to change representation to array form, or arrow form.

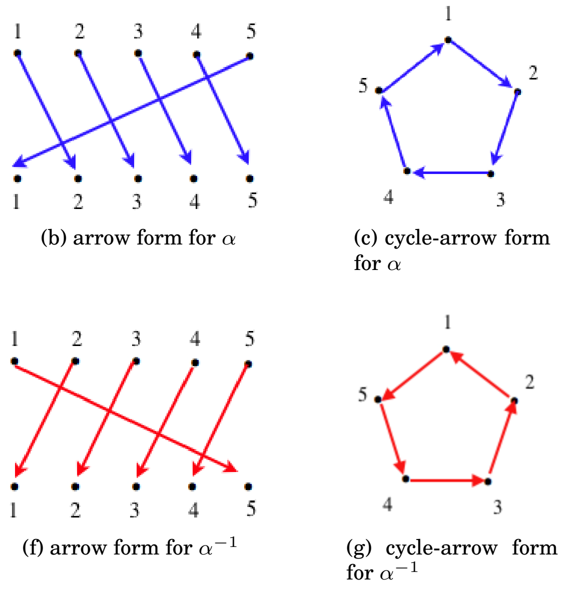
We know that if α is in array form: $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$ then it is straightforward to write down the inverse: $\alpha^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 3 & 1 \end{pmatrix}$. Expressing this in cycle form we have $\alpha^{-1} = (1,5,4,3,2)$. An alternative way to write this cycle is $(5,4,3,2,1)$. This gives us a very simple method for computing an inverse of a cycle: just write the cycle backwards!

$$\alpha^{-1} = (1,2,3,4,5)^{-1} = (5,4,3,2,1) = (1,5,4,3,2)$$

The last equality follows from our convention that we start the cycle with the smallest number in the cycle. Figure 4.2 shows the various representation of α and α^{-1}.

$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$$

(a) array form



(b) arrow form for α

(c) cycle-arrow form for α

$\alpha = (1,2,3,4,5)$

(d) cycle form for α

$$\alpha^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 3 & 4 \end{pmatrix}$$

(e) array form

(f) arrow form for α^{-1}

(g) cycle-arrow form for α^{-1}

$\alpha^{-1} = (1,5,4,3,2)$

(h) cycle form for α^{-1}

Figure 4.2: Different representations for α and α^{-1}.

To make sure we nail this down, consider another example. The inverse of the permutation $\beta = (1,5,3)(2,4)$ is $\beta^{-1} = (2,4)^{-1}(1,5,3)^{-1} = (4,2)(3,5,1) = (2,4)(1,3,5)$.

To summarize:

> To get from the cycle form of α to the cycle form of α^{-1}, just write the representation for α down in the reverse order.

This means, reverse the order in which the numbers are written in each individual cycle, as well as reverse the order in which the cycles are written. Of course, this last step isn't necessary if the cycles are disjoint, since disjoint cycles commute (Theorem 4.1).

Example 4.4: (a) The inverse of the permutation $\alpha = (1,6,3,4,5)$ is $\alpha^{-1} = (5,4,3,6,1) = (1,5,4,3,6)$.

(b) The inverse of a 2-cycle is itself. For example, $(1,2)^{-1} = (2,1) = (1,2)$.

(c) The inverse of the permutation $\beta = (1,4,3,5)(3,7,6)(2,5,7,3,1)(6,4)(2,3,5,4)(4,5,3)$ is

$$\begin{aligned}\beta^{-1} &= [(1,4,3,5)(3,7,6)(2,5,7,3,1)(6,4)(2,3,5,4)(4,5,3)]^{-1}\\ &= (4,5,3)^{-1}(2,3,5,4)^{-1}(6,4)^{-1}(2,5,7,3,1)^{-1}(3,7,6)^{-1}(1,4,3,5)^{-1}\\ &= (4,3,5)(2,4,5,3)(6,4)(2,1,3,7,5)(3,6,7)(1,5,3,4)\end{aligned}$$

Since β^{-1} is not in disjoint cycle form (due to the fact that β itself was not), then we should probably put it in this form.

$$\beta^{-1} = (1,6)(2,7,3,4,5).$$

Exercise 4.4: Let $\alpha = (1,2)(4,5)$ and $\beta = (1,6,5,3,2)$. Compute (a) α^{-1}, (b) β^{-1}, (c) $(\beta\alpha)^{-1}$.

Answer on page 49

4.6 Summary of Permutations

Let's continue with our summary of what we know about S_n.

- S_n, the symmetric group of degree n, is the set of all permutation of $[n] = \{1,2,\dots,n\}$:

$$S_n = \{\alpha \mid \alpha : [n] \to [n] \text{ and } \alpha \text{ is a bijection } \}.$$

- $|S_n| = n!$
- Two elements $\alpha, \beta \in S_n$ can be composed (multiplied) to give another element $\alpha\beta \in S_n$.[2]
- The *identity* permutation $\varepsilon = (1)(2)(3)\cdots(n)$ has the property that $\varepsilon\alpha = \varepsilon\alpha = \alpha$ for all $\alpha \in S_n$. If we follow our convention of omitting 1-cycles, then when writing the cycle form for ε we cannot omit all of them! In this case, we usually write just one 1-cycle. For example, $\varepsilon = (1)$.
- Every $\alpha \in S_n$ has an inverse denoted by α^{-1}. The defining property of an inverse is $\alpha\alpha^{-1} = \alpha^{-1}\alpha = \varepsilon$.
- Inverse of a product: $(\alpha_1\alpha_2\cdots\alpha_k)^{-1} = \alpha_k^{-1}\cdots\alpha_2^{-1}\alpha_1^{-1}$.
- Inverse of an m-cycle: $(a_1, a_2, \dots, a_{m-1}, a_m)^{-1} = (a_m, a_{m-1}, \dots, a_2, a_1)$.
- Permutation composition (multiplication) is associative: $(\alpha\beta)\gamma = \alpha(\beta\gamma) = \alpha\beta\gamma$.
- Permutation composition (multiplication) is not necessarily commutative. However, disjoint permutations commute.
- Cancellation Property: $\alpha\beta = \alpha\gamma$ implies $\beta = \gamma$, and $\beta\alpha = \gamma\alpha$ implies $\beta = \gamma$.
- For every $\alpha \in S_n$ there is a smallest number m, called the order of α, denoted by $\mathrm{ord}(\alpha)$, such that $\alpha^m = \varepsilon$. If a permutation is written in disjoint cycle form then $\mathrm{ord}(\alpha)$ is the least common multiple of the lengths of the cycles.
- We've seen 5 ways to represent a permutation: (1) listing out all the values, (2) array form, (3) arrow form, (4) cycle-arrow form, and (5) cycle form. We will most frequently use cycle form since it is not only the most compact form, it also allows for easy calculations of products, inverses, and orders. We will see very soon that there are many more benefits to this notation.

[2] The convention of these notes is to compose permutations from left-to-right,

4.7 Working with Permutations in Sage

Sage uses disjoint cycle notation for permutations, and permutation composition occurs left-to-right, which agrees with our convention. There are two ways to write the permutation $\alpha = (13)(254)$:
1. As a text string (include quotes): "(1,3)(2,5,4)"
2. As a list of tuples: [(1,3), (2,5,4)]

Sage

```
sage: S5=SymmetricGroup(5)    # symmetric group on 5 objects, and names it S5
sage: a=S5("(2,3)(1,4)")      # constructs the permutation (2,3)(1,4) in S5
sage: b=S5("")                # constructs the identity permutation in S5
sage: c=S5("(2,5,3)")         # constructs the 3-cycle (2,5,3) in S5
sage: print a, b, c,
(1,4)(2,3)
()
(2,5,3)
sage: a*c              # compose permutations by using multiplication sign
(1,4)(3,5)
sage: c.inverse()      # computes inverse
(2,3,5)
sage: c.order()        # computes order
3
```

Try these examples in Sage, then change the examples and see what happens. Don't be afraid to experiment, this is how you learn. You won't break anything (at least it is unlikely you will).

4.8 Exercises

1. **Converting from array to cycle notation.** Convert each of the following permutations given in array form to cycle form

   (a) $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}$

   (b) $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 8 & 5 & 4 & 7 & 1 & 3 & 6 & 2 & 10 & 9 \end{pmatrix}$

   (c) $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 10 & 11 & 9 & 4 & 8 & 15 & 5 & 2 & 7 & 3 & 6 & 1 & 12 & 13 & 14 \end{pmatrix}$

2. **Converting from cycle to array notation.** For each of the following permutation in S_8 convert from cycle form to array form.

   (a) $(1,5,2)(3,4)(7,8)$ (b) $(1,7,4,6)(3,5,8)$ (c) $(1,2)$

3. **Reducing cycle notation to disjoint cycles.**
   When multiplying permutations we will most likely end up with a product of cycles which are not necessarily disjoint, and our goal will be to find a representation in disjoint cycle form. To practice this, write the following permutations in disjoint cycle form.

   (a) $\alpha = (1,4,3,5)(3,7,6)(2,5,7,3,1)(6,4)(2,3,5,4)(4,5,3)$

   (b) $\beta = (1,2,3)(1,4,5)(1,6,7)(1,8,9)$

(c) $\gamma = (9,3,5,6)(4,5,2,3,7)(3,7,8,2)(1,4)(7,4)$

4. **Products and inverses of permutations.**
Consider the following permutations in S_{10}:

$$\alpha = (1,5,2,7)(3,4)(8,10,9), \quad \beta = (1,10,9,7,6,5,2,4,8),$$
$$\gamma = (1,2,3,4)(6,10,8,7,9), \quad \delta = (1,5,8,4)(2,9,10,7)(3,6).$$

Compute the disjoint cycle form of each of the following:

(a) $\alpha\beta$
(b) $\beta\delta$
(c) $\gamma\alpha$
(d) δ^4
(e) $\alpha\gamma\delta$
(f) α^{-1}
(g) $\delta^{-1}\beta^{-1}$
(h) $(\alpha\delta)^{-1}$

5. For each of the permutations below, determine its order.

(a) $\sigma = (3,7,4)$

(b) $\alpha = (1,5,8,4)(2,9,10,7)(3,6)$

(c) $\beta = (2,6,8,3,10,9,7,4)$

(d) $\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 7 & 1 & 5 & 4 & 3 & 6 \end{pmatrix}$

(e) $\delta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 2 & 3 & 4 & 1 & 5 & 10 & 9 & 7 & 6 & 8 \end{pmatrix}$

6. For each of the permutations below, express the inverse in disjoint cycle form.

(a) $\alpha = (1,5,8,4)(2,9,10,7)(3,6)$
(b) $\beta = (2,6,8,3,10,9,7,4)$
(c) $\gamma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 7 & 1 & 5 & 4 & 3 & 6 \end{pmatrix}$
(d) $\delta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 2 & 3 & 4 & 1 & 5 & 10 & 9 & 7 & 6 & 8 \end{pmatrix}$

7. Let $\alpha = (1,3,6)(2,4)$ and $\beta = (1,4,5,2)$. Compute each of the following.
(a) α^{-1} (b) β^{-1} (c) $\alpha\beta$ (d) $\beta\alpha$

8. Let $\alpha = (1,2)(4,5)$ and $\beta = (1,6,5,3,2)$. Compute $\beta^{-1}\alpha\beta$.

9. Show that the order of a m-cycle $(a_1, a_2, \dots, a_m)$ is m.

10. What is the order of a pair of disjoint cycles of length 5 and 3? 4 and 6? 22 and 18?

11. What is the order of the product of three disjoint cycles of lengths 3, 5, and 7? 6, 12 and 26?

12. Show S_5 contains no element of order 7.

13. What is the maximum order of any element in S_{10}?

14. Let $\alpha, \beta \in S_n$, show that α and $\beta^{-1}\alpha\beta$ have the same order.

15. Let $\beta = (1,3,5,7,9,8,6)(2,4,10)$. What is the smallest positive integer n for which $\beta^n = \beta^{-7}$?

16. Let $\alpha = (1,7,4,5,9)(3,8)(10,6,2)$. If α^m is a 5-cycle, what can you say about m?

17. In S_3, find permutations α and β so that ord$(\alpha) = 2$, ord$(\beta) = 2$, and ord$(\alpha\beta) = 3$.

18. Find permutations α and β so that ord$(\alpha) = 3$, ord$(\beta) = 3$, and ord$(\alpha\beta) = 5$.

19. (a) If $\alpha \in S_n$ has order k, show that $\alpha^{-1} = \alpha^{k-1}$.
(b) Use part (a) to find α^{11} for $\alpha = (1,3,6,2)(4,7,5)$.

20. How many permutations of order 5 are there in S_6?

21. Suppose α is a 10 cycle. For which integers i between 2 and 10 is α^i also a 10-cycle?

22. **Splicing and dicing cycles.**[3] What happens to the cycle structure of a permutation α when you

[3] This exercise is from J. Kiltinen's book *Oval Track and Other Permutation Puzzles*.

follow α by a transposition? The answer is you either splice two of the cycles of α into one bigger cycle, you cut one of the cycles of α into two smaller cycles, you extend one cycle by on element, or you add a new transposition to the cycle structure. Verify the special cases of this statement below, and then make an argument that the claim follows in general from these special cases.

(a) If $\alpha = (a_1, a_2, \ldots, a_r)(b_1, b_2, \ldots, b_s)$ where these two cycles are disjoint, then

$$\alpha(a_1, b_1) = (a_1, \ldots, a_r, b_1, \ldots, b_s).$$

(b) If $\beta = (a_1, a_2, \ldots, a_r)$ and $1 \leq i < j \leq r$, then

$$\beta(a_i, a_j) = (a_1, \ldots, a_{i-1}, a_j, a_{j+1}, \ldots, a_r)(a_i, a_{i+1}, \ldots, a_{j-1}).$$

(c) If $\gamma = (a_1, a_2, \ldots, a_r)$ and $b \neq a_i$ for all i, then

$$\gamma(a_1, b) = (a_1, a_2, \ldots, a_r, b).$$

(d) If $\delta = (a_1, a_2, \ldots, a_r)$ and if (b_1, b_2) is disjoint from δ, then

$$\delta(b_1, b_2) = (a_1, a_2, \ldots, a_r)(b_1, b_2).$$

Answers to in-chapter exercises:

Exercise 4.1: $(1, 3)(2, 4)$

Exercise 4.2: $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 1 & 5 & 7 & 2 & 6 & 4 & 8 \end{pmatrix}$

Exercise 4.3: (a) 2 (b) 4 (c) 12

Exercise 4.4: (a) $(1, 2)(4, 5)$ (b) $(1, 2, 3, 5, 6)$ (c) $(1, 3, 5, 4, 6)$

Lecture 5

From Puzzles To Permutations

5.1 Introduction

The puzzles we have encountered so far all have a common theme: the pieces can be mixed up, and the goal is to restore the pieces back to some proper order. In other words, the pieces are permuted. We have already introduced, from a mathematical viewpoint, the notion of a permutation: *A permutation of a set A is defined to be a bijection $\alpha : A \to A$.* We will now see how to represent a puzzle by a permutation.

There are two types of permutations associated with a puzzle:

(a) the permutation describing the puzzles current position,

(b) the permutation corresponding to a move sequence applied to the puzzle.

Note that the second one can really be thought of as a special case of the first. This is because the permutation we assign to a move sequence is just the one which represents the puzzle position after the move is applied to the solved state.

In this section we discuss how to write down these permutations for the standard set of puzzles we are studying. If we are thoughtful in how we do this, then the permutation describing the puzzles position is a composition of the permutations corresponding to the puzzle moves which takes the puzzle from the solved state to that position. That is, multiplying the permutations corresponding to the *moves*, should give us the permutation of the resulting *position*.

The way we do this will be the same for all puzzles where both the moving pieces and home positions have been labelled by numbers in $[n]$.

Definition 5.1 (Puzzle Position $\to$ Permutation): For a given position (scrambling) of the puzzle, the **permutation corresponding to this position** is $\alpha : S_n \to S_n$ where

$$\alpha(i) = j \quad \text{if the piece labelled } i \text{ is in the position labelled } j.$$

This permutation describes precisely how the pieces in the home (or solved) state configuration were moved to produce the current configuration.

Definition 5.2 (Puzzle Move $\rightarrow$ Permutation): For a given move sequence applied to the puzzle, the **permutation corresponding to this move sequence** is $\beta : S_n \rightarrow S_n$ where

$$\beta(i) = j \quad \text{if the piece in position labelled } i \text{ moved to position labelled } j.$$

These definitions show how to construct the function α which corresponds to a position/move, but we should really say a few words about why this function is actually a permutation. That is, we want to observe α is one-to-one and onto. To see this, notice in any scrambling of the pieces no position has more than one piece occupying it, in other words, distinct pieces have gone into distinct positions. This means α is a one-to-one map from $[n]$ to $[n]$, which is then necessarily onto. This observation suggests why α is indeed a permutation.

Theorem 5.1 (Multiplying Moves): *Let α be the permutation corresponding to the current position of the puzzle, and β_1, β_2, $\ldots$ β_k be a move sequence applied to the puzzle which results in a final position γ. Then*

$$\alpha\beta_1\beta_2\cdots\beta_k = \gamma.$$

Proof: To see why this is true, consider any piece of the puzzle, say the piece labelled ℓ. Then, before the move sequence is applied, the piece ℓ starts in position $x_0 = \alpha(\ell)$. As the moves are applied one-by-one the ℓ piece moves to position x_1, then to position x_2, and so on, until it finally ends up in position x_k, where

$$\begin{aligned}
x_1 &= \beta_1(x_0) = \beta_1(\alpha(\ell)) = (\alpha\beta_1)(\ell) \\
x_2 &= \beta_2(x_1) = \beta_2((\alpha\beta_1)(\ell)) = (\alpha\beta_1\beta_2)(\ell) \\
&\vdots \\
x_k &= \beta_k(x_{k-1}) = \beta_k((\alpha\beta_1\beta_2\cdots\beta_{k-1})(\ell)) = (\alpha\beta_1\beta_2\cdots\beta_k)(\ell)
\end{aligned}$$

Therefore, $\gamma(\ell) = x_k = (\alpha\beta_1\beta_2\cdots\beta_k)(\ell)$ for every $\ell \in [n]$, and so $\gamma = \alpha\beta_1\beta_2\cdots\beta_k$. This proves the theorem. □

Over the next few sections we will look at each puzzle individually.

5.2 Swap

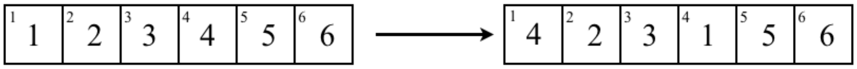
Each arrangement of the numbers in the Swap Puzzle, say with n numbers, is a permutation of the set $[n] = \{1, 2, 3, \ldots, n\}$. For example, consider the following position of Swap with 6 numbers.

1	2	3	4	5	6
4	6	1	2	3	5

The permutation $\alpha : [6] \rightarrow [6]$ we associate to this position is determined as follows. Since tile number 1 is in box number 3, then $\alpha(1) = 3$. Since tile 2 is in box 4, then $\alpha(2) = 4$. Continuing in this fashion we find α maps numbers 1 through 8 as follows.

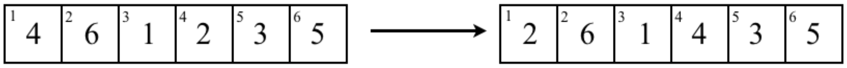
$$\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 1 & 6 & 2 \end{pmatrix} \quad \text{or in cycle form} \quad \alpha = (1, 3, 5, 6, 2, 4).$$

Consider the move obtained by swapping the tiles in boxes 1 and 4. What permutation should we use to represent this move? If we think of applying this move to the solved state of the puzzle, for example:



then we can represent this move by the permutation β corresponding to the position it leaves the puzzle in: $\beta = (1,4)$.

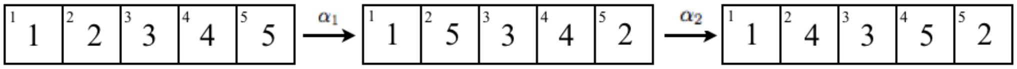
Now imagine the puzzle was not in the solved state, and we were applying the 1, 4 swap. For example, we apply the move as follows:



How should we assign a permutation to this move? Well, the simplest way is to say it is just the same move as above, ignoring the actually objects in the boxes. All that matters is that the contents of box 1 and box 4 were switched. The permutation should then only depend on the boxes involved and how the contents move between boxes, but is shouldn't depend on what exactly is in the boxes. This is the essence of Definition 5.2.

From now on, when we wish to describe a move, we can just state it by giving the corresponding permutation. For example, the permutation $(3,7)$ represents the move of switching the contents of boxes 3 and 7.

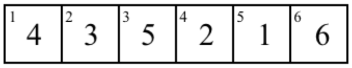
Example 5.1: Consider the following sequence of moves in Swap.



The first move consists of swapping the contents of boxes 2 and 5 so it corresponds to the permutation $\alpha_1 = (2,5)$. The second move consists of swapping the contents of boxes 2 and 4 so it corresponds to the permutation $\alpha_2 = (2,4)$. The product $\alpha_1\alpha_2 = (2,5)(2,4) = (2,5,4)$, which is the permutation representing the move sequence as a whole, is precisely the permutation corresponding to the final position.

Example 5.2: Apply the move sequence $\tau_1 = (3,5)$, $\tau_2 = (1,2)$, $\tau_3 = (2,5)$, $\tau_4 = (1,4)$ to the game of Swap with 6 objects, and draw the final position of the game board, assuming you began with it in the solved state.

The move sequence corresponds to the single maneuver: $\alpha = \tau_1\tau_2\tau_3\tau_4 = (3,5)(1,2)(2,5)(1,4) = (1,5,3,2,4)$, (Theorem 5.1) which means the resulting game board position is as follows.

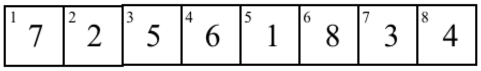


We could have also applied the move sequences one-by-one to achieve the same result (here we simply write the numbered tiles, as they appear on the game board, separated by vertical bars |):

$$1|2|3|4|5|6 \xrightarrow{\tau_1=(3,5)} 1|2|5|4|3|6 \xrightarrow{\tau_2=(1,2)} 2|1|5|4|3|6 \xrightarrow{\tau_3=(2,5)} 2|3|5|4|1|6 \xrightarrow{\tau_4=(1,4)} 4|3|5|2|1|6$$

Example 5.3: Write the permutation $\alpha = (1,5,3,7)(4,8,6)$ as a product of 2-cycles. (Hint: Solve the corresponding Swap puzzle.)

The permutation α corresponds to the position



which we will simply write as 7|2|5|6|1|8|3|4. To solve the puzzle from this state we may do the following:

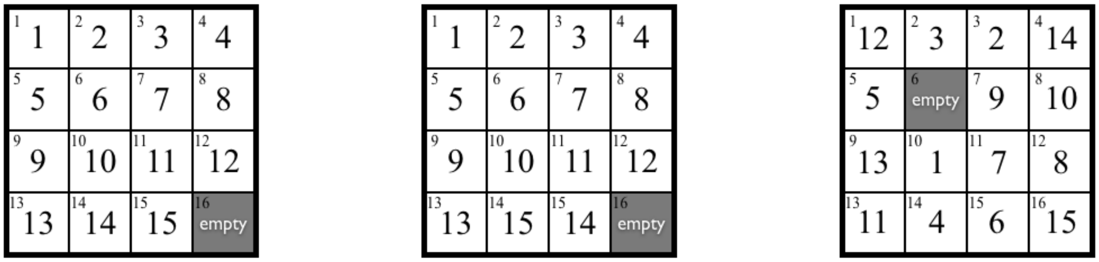
$$\alpha \Rightarrow 7|2|5|6|1|8|3|4 \xrightarrow{\tau_1=(1,5)} 1|2|5|6|7|8|3|4 \xrightarrow{\tau_2=(3,7)} 1|2|3|6|7|8|5|4 \xrightarrow{\tau_3=(4,8)} 1|2|3|4|7|8|5|6$$
$$\xrightarrow{\tau_4=(5,7)} 1|2|3|4|5|8|7|6 \xrightarrow{\tau_5=(6,8)} 1|2|3|4|5|6|7|8 \Rightarrow \varepsilon.$$

This means $\alpha\tau_1\tau_2\tau_3\tau_4\tau_5 = \varepsilon$, or $\alpha = \tau_5^{-1}\tau_4^{-1}\tau_3^{-1}\tau_2^{-1}\tau_1^{-1}$. Therefore, we have found a decomposition of α into 2-cycles:

$$\alpha = (1,5,3,7)(4,8,6) = (6,8)(5,7)(4,8)(3,7)(1,5).$$

5.3 15-Puzzle

Imagine the tiles in the 15 puzzle mixed-up. Consider Figure 5.1c for example. Each tile was moved from some numbered box (its home box) to some other numbered box: for example the tile in box 1 moved to box 10, but the tile in box 5 stayed in box 5. Here we think of the empty space as tile number 16, which we will often call the "empty tile".



(a) (b) (c)

Figure 5.1: The 15 Puzzle

We can write down the permutations describing each of the positions in 5.1 by using Definition 5.1.

(a) This puzzle is in the solved state, so no tiles have been moved. This corresponds to the identity permutation ε. The array form of this permutation is

$$\varepsilon = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \end{pmatrix}.$$

(b) In this puzzle the tiles in boxes 14 and 15 were switched. This corresponds to the permutation $(14,15)$. The array form of this permutation is

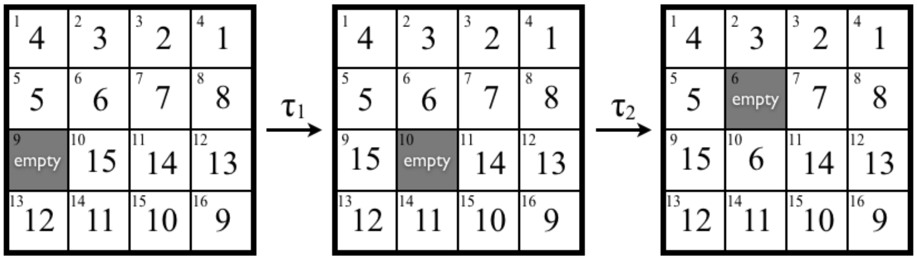
$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 15 & 14 & 16 \end{pmatrix}.$$

(c) Tile 1 is now in box 10, so $1 \mapsto 10$ for this permutation. Tile 10 is in box 8, so $10 \mapsto 8$. Continuing in this fashion we construct the cycle form of the corresponding permutation: $(1,10,8,12)(2,3)(4,14)(6,15,16)(7,11,13,9)$, where we omitted the 1-cycle (5). The array form of this permutation is

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ 10 & 3 & 2 & 14 & 5 & 15 & 11 & 12 & 7 & 8 & 13 & 1 & 9 & 4 & 16 & 6 \end{pmatrix}.$$

By the construction of the 15-Puzzle a legal move consists of swapping a tile with the empty tile, provided it is adjacent to the empty tile. This means legal moves are 2-cycles, and move sequences are products of 2-cycles.

Example 5.4: Consider the following sequence of moves in the 15-Puzzle.



The first move consists of moving the empty space from box 9 to 10 so it corresponds to the permutation $\tau_1 = (9, 10)$. The second move consists of moving the empty space form box 10 to 6 so it corresponds to the permutation $\tau_2 = (10, 6)$.

The first position is given by $\alpha = (1, 4)(2, 3)(9, 16)(15, 10)(11, 14)(12, 13)$, the last position is $\beta = (1, 4)(2, 3)(6, 10, 15, 9, 16)(11, 14)(12, 13)$, and we have

$$\begin{aligned} \alpha\tau_1\tau_2 &= (1, 4)(2, 3)(9, 16)(15, 10)(11, 14)(12, 13)(9, 10)(10, 6) \\ &= (1, 4)(2, 3)(6, 10, 15, 9, 16)(11, 14)(12, 13) \\ &= \beta. \end{aligned}$$

This provides an illustration of Theorem 5.1.

5.4 Oval Track Puzzle

Since there are 20 moving disks on the Oval Track puzzle (Figure 5.2), each position/move can be described as a permutation of $[20] = \{1, 2, \dots, 20\}$.

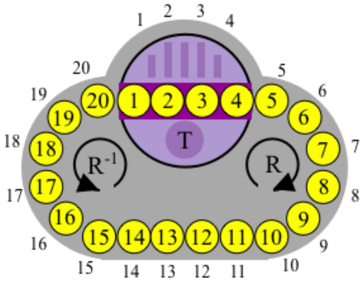


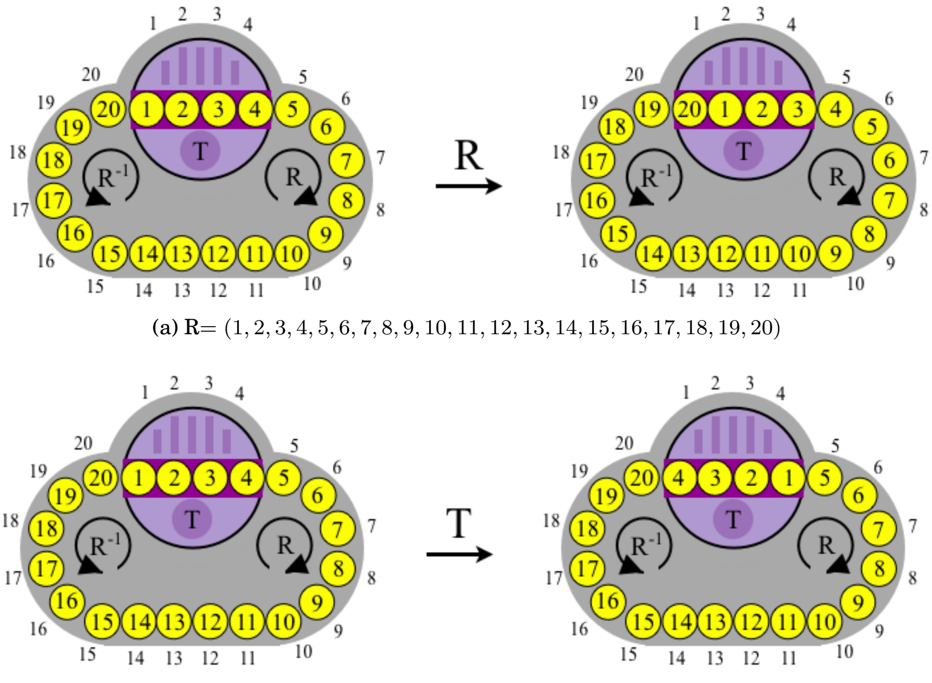
Figure 5.2: The Oval Track Puzzle.

Recall from Lecture 1, the basic legal moves of the oval track puzzle are R, R^{-1}, and T. Where R denotes a clockwise rotation of numbers around the track, moving each number one space, R^{-1}denotes a counterclockwise rotation of the numbers around the track, and T denotes a rotation of the turntable. See Figure 5.3.

The permutation corresponding to the legal moves R, R^{-1}, and T are as follows:

$$\begin{aligned} &R = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20) \\ &R^{-1} = (1, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2) \\ &T = (1, 4)(2, 3) \end{aligned}$$

Note that $T^{-1} = T$. This is due to the fact that spinning the turntable in either direction achieves the same result.

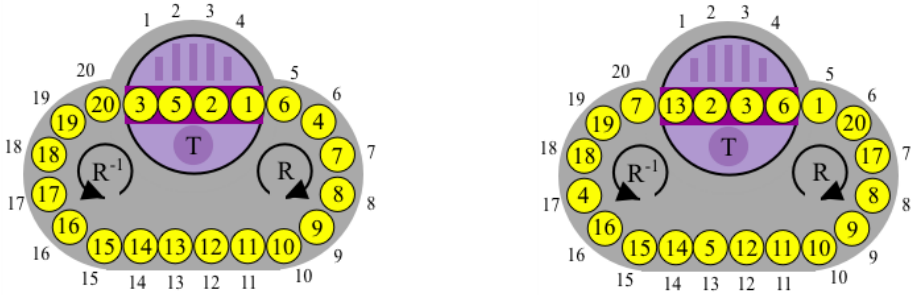


(a) R= $(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20)$

(b) T $= (1, 4)(2, 3)$

Figure 5.3: Basic Moves R and T of Oval Track.

Example 5.5: Express, in cycle form, the permutations describing each of the positions in Figure 5.4.



(a)

(b)

Figure 5.4: Oval Track scramblings for Example 5.5.

(a) Disk 1 is in slot 4, disk 4 is in slot 6, disk 6 is in slot 5, disk 5 is in slot 2, disk 2 is in slot 3, and disk 3 is in slot 1. All other disks are still in their home positions, so the corresponding permutation is $(1, 4, 6, 5, 2, 3)$.

(b) Similar to part (a) we just follow where each disk ended up. The corresponding permutation is $(1, 5, 13)(4, 17, 7, 20, 6)$.

Example 5.6: Apply each of the following move sequences to the solved-state Oval Track puzzle and draw the resulting configuration of the disks on the puzzle.

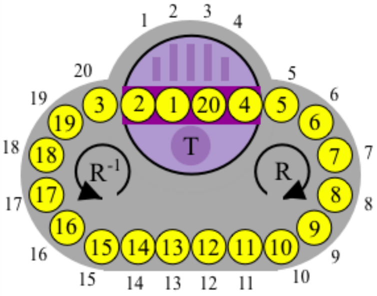
(a) RTR^{-1}

(b) $R^{-4}TR^{2}TR^{-1}$

(a) If you have a physical puzzle, or one of the virtual ones linked to from the course website, then you can actually perform the move sequence and attain the resulting configuration. We can also do this using the permutation representations of the move sequence:

$$\begin{aligned}RTR^{-1} &= (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)(1,4)(2,3)R^{-1}\\ &= (1,3)(4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)R^{-1}\\ &= (1,3)(4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)\\ &\qquad (1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)\\ &= (1,2)(3,20)\end{aligned}$$

The resulting position is drawn below.

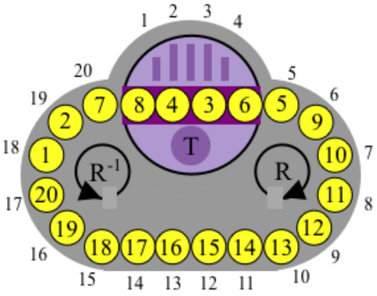


(b) Multiplying two 20-cycles and a 4-cycle in part (a) was not technically difficult, it was just tedious. This product $R^{-4}TR^{2}TR^{-1}$ would be very tedious, and the actual calculation wouldn't be too enlightening. No mathematician would actually do the calculation by hand. In fact, most would not have done part (a) by hand either. It is really just the end result we are interested in, so we should do what any normal person would do, have a computer do the calculation. We'll use Sage to do this.

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4)(2,3)")
sage: R^(-4)*T*R^(2)*T*R^(-1)
(1,18,15,12,9,6,4,2,19,16,13,10,7,20,17,14,11,8)
```

Later on we will discuss in detail the commands used above, and what each line of code does (you may be able to figure this out for yourself). For now, we have our answer to the question, $R^{-4}TR^{2}TR^{-1}$ corresponds to the permutation returned by Sage and the puzzle looks like this



5.5 Hungarian Rings

There are 38 moving disks in the the (numbered) Hungarian Rings puzzle (Figure 5.5), so each position/move can be described as a permutation of $[38] = \{1, 2, \ldots, 38\}$.

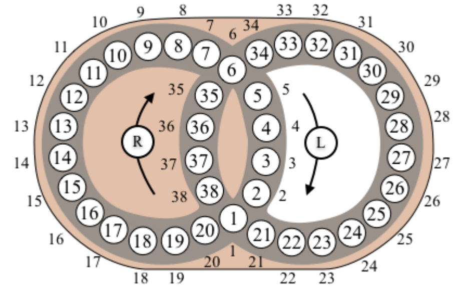


Figure 5.5: Hungarian Rings - numbered version.

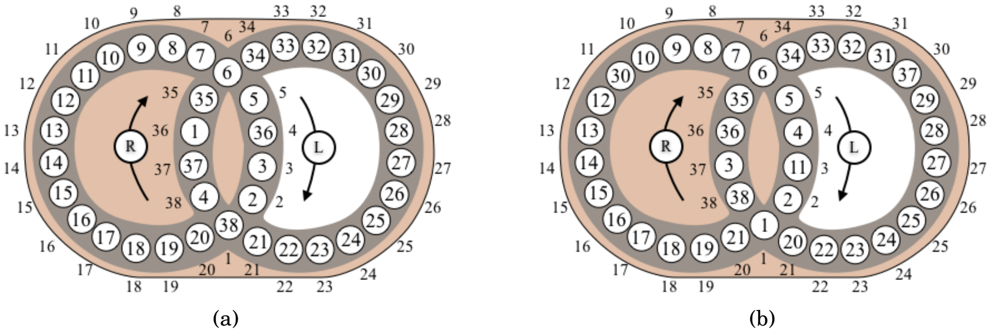
Recall from Lecture 1, the basic legal moves of the Hungarian Rings puzzle are R, R^{-1}, L, and L^{-1}, where R denotes a clockwise rotation of numbers around the right-hand ring (each number moves one space), R^{-1}denotes a counterclockwise rotation of the numbers around the right-hand ring, L denotes a clockwise rotation of numbers around the left-hand ring, and L^{-1}denotes a counterclockwise rotation of the numbers around the left-hand ring.

The permutation corresponding to each of the legal moves R and L are:

$$R = (1, 38, 37, 36, 35, 6, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21)$$
$$L = (1, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2)$$

$\mathrm{R}^{-1}$and L^{-1}correspond to the inverses of these permutations.

Example 5.7: Express, in cycle form, the permutations describing each of the positions in Figure 5.6.



(a)

(b)

Figure 5.6: Hungarian Rings scramblings.

(a) We simply follow where each disk has been moved. The corresponding permutation is $(1, 36, 4, 38)$.

(b) Following where each disk has been moved, the corresponding permutation is $(3, 37, 30, 11)(20, 21)$.

Example 5.8: For each of the following move sequences, which were applied to the solved-state Hungarian Rings puzzle, draw the resulting configuration of the disks on the puzzle.

(a) $R^{-1}LR$

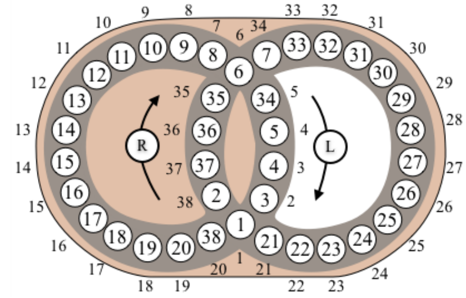
(b) $L^5R^5L^{-5}R^{-5}$

(a) If you have a physical puzzle, or one of the virtual ones linked to from the course website, then you can actually perform the move sequence and attain the resulting configuration. We can also do this using the permutation representations of the move sequence, by multiplying the permutations. We'll use Sage to do the computations.

Sage

```
sage: S38=SymmetricGroup(38)
sage: L=S38("(1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)")
sage: R=S38("(1,38,37,36,35,6,34,33,32,31,30,29,28,27,26,25,24,23,22,21)")
sage: R^(-1)*L*R
(2,38,20,19,18,17,16,15,14,13,12,11,10,9,8,7,34,5,4,3)
```

The resulting position is drawn below.

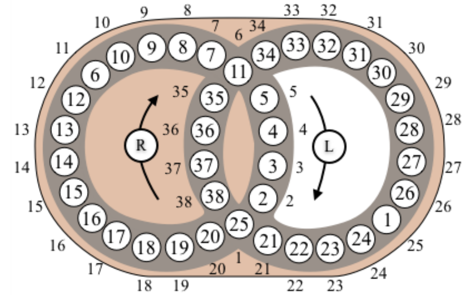


(b) Again, we'll use Sage to do the desired calculation.

Sage

```
sage:  L^5*R^5*L^(-5)*R^(-5)
(1,25)(6,11)
```

The resulting position is drawn below.



5.6 Rubik's Cube

To keep track of how the pieces of the cube move around, and to be able to describe movements and positions by permutations, we label each of the facets with numbers. For the $2 \times 2 \times 2$ cube there are 24 facets, whereas for the $3 \times 3 \times 3$ cube there are 54, but only 48 actually move. The 6 centres can be thought of as remaining fixed (though they can rotate, but this is only noticeable if the sticker has an image on it).

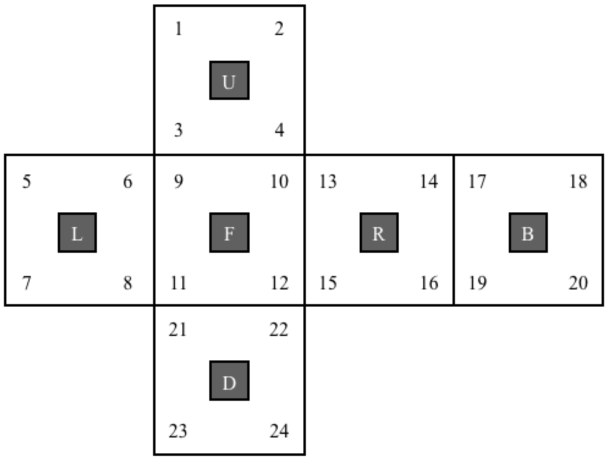
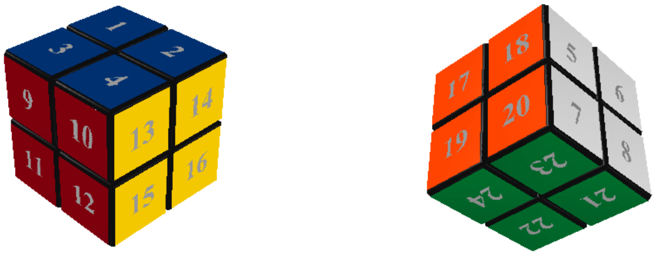


Figure 5.7: Facet labeling on the Pocket cube.



(a) Labeling on Up (blue), Right (yellow), Front (red) faces

(b) Labeling on Down (green), Back (orange), Left (white) faces

Figure 5.8: The labeling of the facets of the Pocket Cube.

5.6.1 $2 \times 2 \times 2$ Cube

We label the facets of the Pocket Cube as shown in Figure 5.7. Figure 5.8 shows the labeling on an actual cube.

We associate permutations to positions and moves in the usual way (Definitions 5.1 and 5.2). The basic moves of the Rubik's Cube are R, L, U, D, F, B, and their inverses. Each one denotes a clockwise quarter turn of the corresponding face. See Lecture 1 for a thorough discussion of this notation.

The permutation corresponding to each of the basic moves of the Pocket Cube are:

$$\begin{aligned}
\mathrm{R} &= (13,14,16,15)(10,2,19,22)(12,4,17,24)\\
\mathrm{L} &= (5,6,8,7)(3,11,23,18)(1,9,21,20)\\
\mathrm{U} &= (1,2,4,3)(9,5,17,13)(10,6,18,14)\\
\mathrm{D} &= (21,22,24,23)(11,15,19,7)(12,16,20,8)\\
\mathrm{F} &= (9,10,12,11)(3,13,22,8)(4,15,21,6)\\
\mathrm{B} &= (17,18,20,19)(1,7,24,14)(2,5,23,16)
\end{aligned}$$

R^{-1}, L^{-1}, U^{-1}, D^{-1}, F^{-1}, D^{-1}correspond to the inverses of these permutations.

5.6.2 $3 \times 3 \times 3$ Cube

As we described in Lecture 1, we label the facets of the Rubik's Cube as shown Figure 5.9. Figure 5.10 shows the labeling on an actual cube.

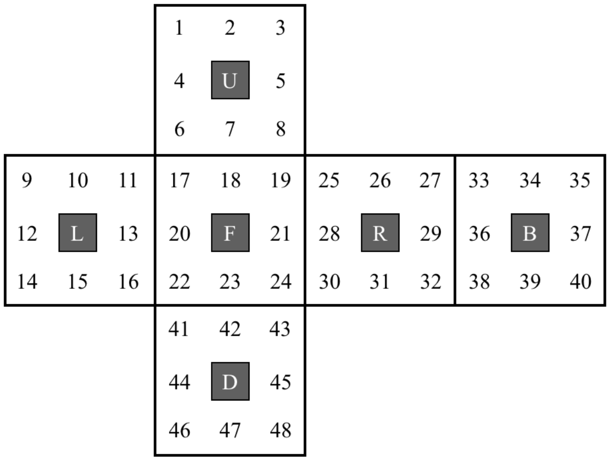
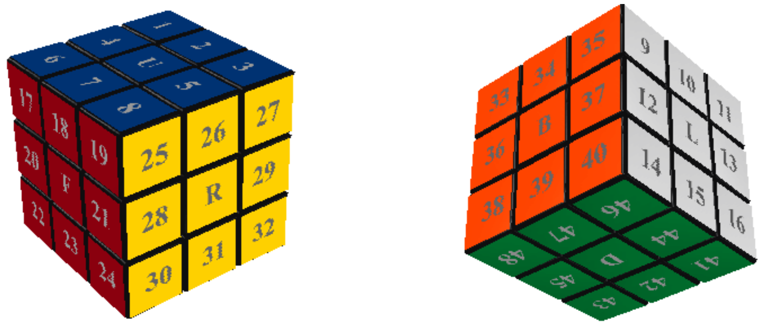


Figure 5.9: Facet labeling on the Rubik's cube.


1 2 3 4 U 5 6 7 8
17 18 19 20 F 21 22 23 24
25 26 27 28 R 29 30 31 32
33 34 35 36 B 37 38 39 40
9 10 11 12 L 13 14 15 16
41 42 43 44 D 45 46 47 48

(a) Labeling on Up, Right, Front faces

(b) Labeling on Down, Back, Left faces

Figure 5.10: The labeling of the facets of Rubik's Cube.

The permutation corresponding to each of the basic moves of the Rubik's Cube are:

$$
\begin{aligned}
&\mathrm{R} = (25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)\\
&\mathrm{L} = (9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)\\
&\mathrm{U} = (1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)\\
&\mathrm{D} = (41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)\\
&\mathrm{F} = (17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)\\
&\mathrm{B} = (33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)
\end{aligned}
$$

R^{-1}, L^{-1}, U^{-1}, D^{-1}, F^{-1}, D^{-1}correspond to the inverses of these permutations.

Uncovering the secrets of the cube will involve playing around with these permutations, and our playground will be Sage . Here is how we can input the permutations for the Rubik's Cube into Sage.

```
Sage
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
```

```
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
```

We could then, for instance, see what the move sequence RU does to the cube.

Sage

```
sage: R*U
(1,3,38,43,11,35,27,32,30,17,9,33,48,24,6)(2,5,36,45,21,7,4)(8,25,19)  \
(10,34,26,29,31,28,18)
```

We can easily "eyeball" the order of this permutation, it is $\text{lcm}(15, 7, 3) = 105$. This gives us a glimpse into the kind of questions we can easily answer through computation.

Also, since RU consists of a 15-cycle, a 3-cycle and two 7-cycles, raising it to the power of 15 would get rid of the 15- and 3-cycles, and would leave us with some 7-cycles.

Sage

```
sage: (R*U)^15
(2,5,36,45,21,7,4)(10,34,26,29,31,28,18)
```

This means we can move fewer pieces by taking powers of some move sequences. We'll later how this is an effective strategy for solving these puzzles.

5.7 Exercises

1. **Swap Puzzle arrangements into cycle notation.** For each of the following scramblings of the tiles in Swap, express them as permutations in S_n using cycle notation.

(a)

1	2	3
2	3	1

(b)

1	2	3	4	5
1	4	5	2	3

(c)

1	2	3	4	5	6	7	8	9	10
1	8	3	2	5	4	7	6	9	10

(d)

1	2	3	4	5	6	7	8	9	10
9	5	10	6	2	1	3	8	4	7

(e)

1	2	3	4	5	6	7	8	9	10	11	12
4	6	9	12	8	10	1	7	2	5	3	11

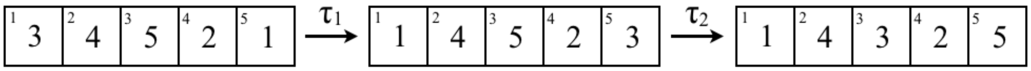
2. **Swap Puzzle arrangements from cycle notation.** For each of the following permutations, given in cycle form, draw the corresponding scrambling of the tiles on the Swap puzzle.

(a) $(1, 5, 3, 8)(2, 4, 7)$

(b) $(3, 7, 4, 10, 6, 5, 8)$

(c) $(1, 12)(2, 11)(3, 10)(5, 6, 7)$

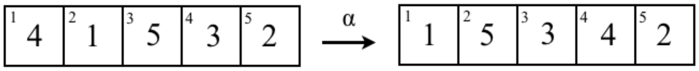
3. **Swap Puzzle arrangements and moves in cycle notation.** In each part (a) - (c) below, a sequence of moves has been applied to a scrambling of the tiles in Swap. Do the following:
(i) Express the starting position α as a permutation in cycle notation.
(ii) Express each move τ_i as a 2-cycle.
(iii) Express the whole move sequence as a permutation in cycle notation.
(iv) Express the final position β as a permutation in cycle notation and show that $\alpha\tau_1 \cdots \tau_n = \beta$.

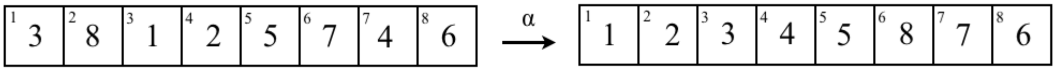
   (a) 

   (b) 5|4|2|8|1|3|6|7 $\xrightarrow{\tau_1}$ 5|2|4|8|1|3|6|7 $\xrightarrow{\tau_2}$ 8|2|4|5|1|3|6|7 $\xrightarrow{\tau_3}$ 1|2|4|5|8|3|6|7 $\xrightarrow{\tau_4}$ 1|2|4|5|8|6|3|7

   (c) 5|10|4|1|6|7|2|3|8|9 $\xrightarrow{\tau_1}$ 5|10|4|1|7|6|2|3|8|9 $\xrightarrow{\tau_2}$ 1|10|4|5|7|6|2|3|8|9 $\xrightarrow{\tau_3}$ 1|9|4|5|7|6|2|3|8|10 $\xrightarrow{\tau_4}$ 1|9|4|7|5|6|2|3|8|10 $\xrightarrow{\tau_5}$ 1|2|4|7|5|6|9|3|8|10

4. **Swap Puzzle move sequence in cycle notation.** For each move sequence α given below, express it as a permutation in cycle form.

   (a) 

   (b) 

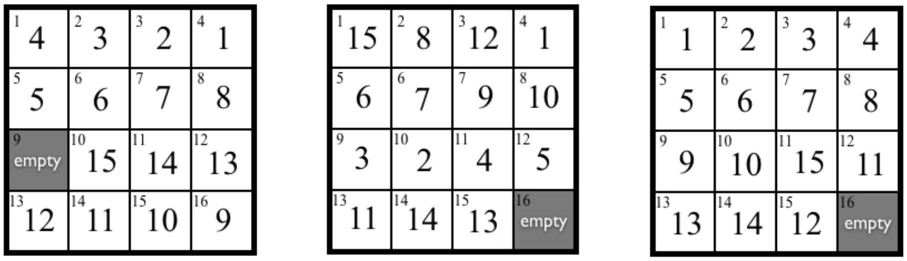
5. **Decomposing a permutation into 2-cycles.** Write the permutation $\alpha = (1, 2, 3)$ as a product of 2-cycles. (Hint: Solve the corresponding Swap puzzle.)

6. **Decomposing a permutation into 2-cycles.** Write the permutation $\alpha = (1, 2, 8, 3, 7)(4, 5, 6)$ as a product of 2-cycles. (Hint: Solve the corresponding Swap puzzle.)

7. **Decomposing a permutation into 3-cycles.** Write the permutation $\alpha = (1, 2)(3, 4)$ as a product of 3-cycles. (Hint: Solve the corresponding Swap puzzle, under the variation where the legal moves are now 3-cycles.)

8. **Decomposing a permutation into 3-cycles.** Write the permutation $\alpha = (1, 2, 8, 3, 7)(4, 5, 6)$ as a product of 3-cycles. (Hint: Solve the corresponding Swap puzzle, under the variation where the legal moves are now 3-cycles.)

9. **15-Puzzle arrangements into cycle notation.** Express each of the following scramblings of the 15-puzzle as a permutation in cycle form.

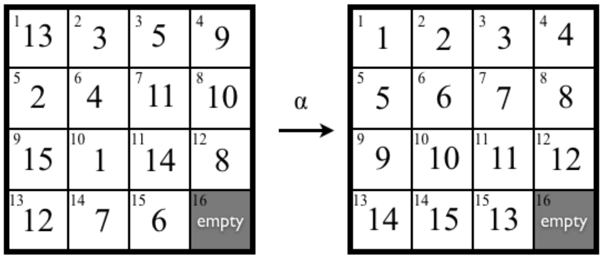


(a) (b) (c)

10. **15-Puzzle arrangements from cycle notation.** For each of the following permutations, given in cycle form, draw the corresponding scrambling of the tiles on the 15 puzzle.

   (a) $(6, 7, 11, 10)$
   (b) $(1, 5, 3, 10, 15, 2, 14, 12, 11, 6, 7, 4)(9, 16)$
   (c) $(2, 10, 13, 5)(1, 3)(7, 8, 9)$

11. **15-Puzzle move sequence in cycle notation.** For the move sequence α given below, express it as a permutation in cycle form.



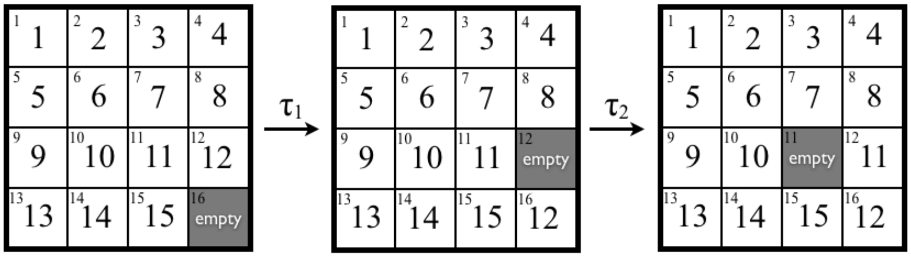
^{1}13	^{2}3	^{3}5	^{4}9
^{5}2	^{6}4	^{7}11	^{8}10
^{9}15	^{10}1	^{11}14	^{12}8
^{13}12	^{14}7	^{15}6	^{16}empty

$\xrightarrow{\alpha}$

^{1}1	^{2}2	^{3}3	^{4}4
^{5}5	^{6}6	^{7}7	^{8}8
^{9}9	^{10}10	^{11}11	^{12}12
^{13}14	^{14}15	^{15}13	^{16}empty

12. **15-Puzzle arrangements and moves in cycle notation.** In each part (a) - (c) below, a sequence of moves has been applied to a scrambling of the tiles in the 15-Puzzle. Do the following:
(i) Express the starting position α as a permutation in cycle notation.
(ii) Express each move τ_i as a 2-cycle.
(iii) Express the whole move sequence as a permutation in cycle notation.
(iv) Express the final position β as a permutation in cycle notation and show that $\alpha\tau_1 \cdots \tau_n = \beta$.

(a)



^{1}1	^{2}2	^{3}3	^{4}4
^{5}5	^{6}6	^{7}7	^{8}8
^{9}9	^{10}10	^{11}11	^{12}12
^{13}13	^{14}14	^{15}15	^{16}empty

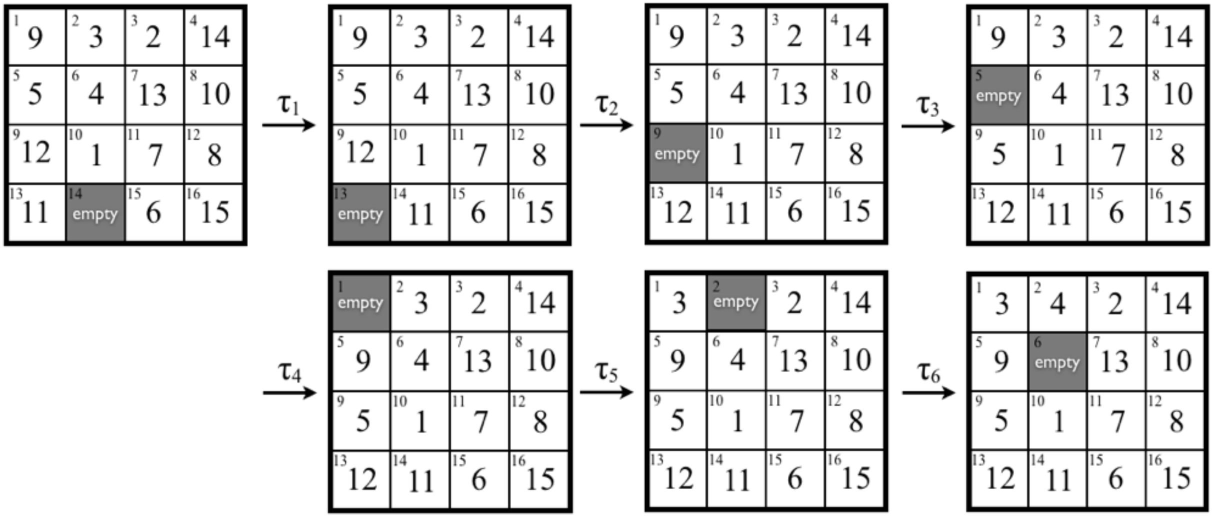
$\xrightarrow{\tau_1}$

^{1}1	^{2}2	^{3}3	^{4}4
^{5}5	^{6}6	^{7}7	^{8}8
^{9}9	^{10}10	^{11}11	^{12}empty
^{13}13	^{14}14	^{15}15	^{16}12

$\xrightarrow{\tau_2}$

^{1}1	^{2}2	^{3}3	^{4}4
^{5}5	^{6}6	^{7}7	^{8}8
^{9}9	^{10}10	^{11}empty	^{12}11
^{13}13	^{14}14	^{15}15	^{16}12

(b)



^{1}9	^{2}3	^{3}2	^{4}14
^{5}5	^{6}4	^{7}13	^{8}10
^{9}12	^{10}1	^{11}7	^{12}8
^{13}11	^{14}empty	^{15}6	^{16}15

$\xrightarrow{\tau_1}$

^{1}9	^{2}3	^{3}2	^{4}14
^{5}5	^{6}4	^{7}13	^{8}10
^{9}12	^{10}1	^{11}7	^{12}8
^{13}empty	^{14}11	^{15}6	^{16}15

$\xrightarrow{\tau_2}$

^{1}9	^{2}3	^{3}2	^{4}14
^{5}5	^{6}4	^{7}13	^{8}10
^{9}empty	^{10}1	^{11}7	^{12}8
^{13}12	^{14}11	^{15}6	^{16}15

$\xrightarrow{\tau_3}$

^{1}9	^{2}3	^{3}2	^{4}14
^{5}empty	^{6}4	^{7}13	^{8}10
^{9}5	^{10}1	^{11}7	^{12}8
^{13}12	^{14}11	^{15}6	^{16}15

$\xrightarrow{\tau_4}$

^{1}empty	^{2}3	^{3}2	^{4}14
^{5}9	^{6}4	^{7}13	^{8}10
^{9}5	^{10}1	^{11}7	^{12}8
^{13}12	^{14}11	^{15}6	^{16}15

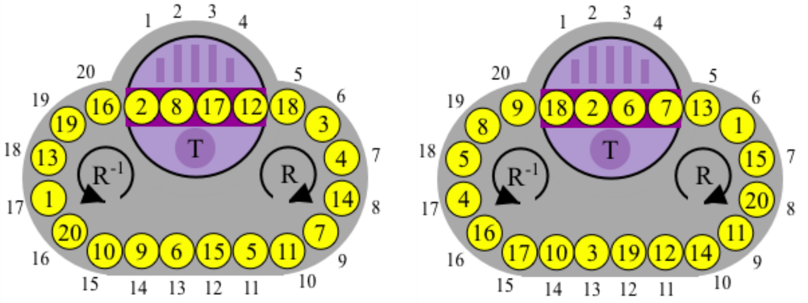
$\xrightarrow{\tau_5}$

^{1}3	^{2}empty	^{3}2	^{4}14
^{5}9	^{6}4	^{7}13	^{8}10
^{9}5	^{10}1	^{11}7	^{12}8
^{13}12	^{14}11	^{15}6	^{16}15

$\xrightarrow{\tau_6}$

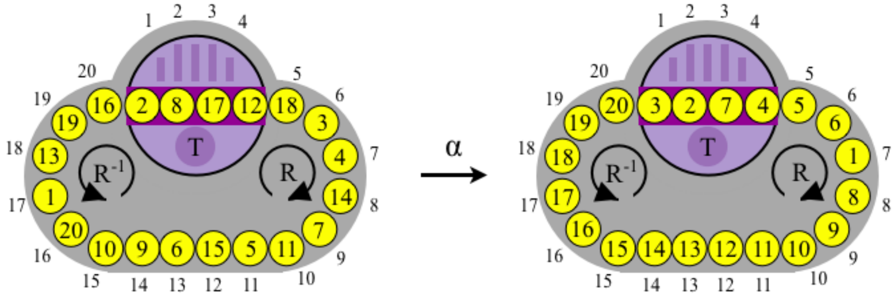
^{1}3	^{2}4	^{3}2	^{4}14
^{5}9	^{6}empty	^{7}13	^{8}10
^{9}5	^{10}1	^{11}7	^{12}8
^{13}12	^{14}11	^{15}6	^{16}15

13. **Oval Track Puzzle arrangements into cycle notation.** Express, in cycle form, the permutation describing each of the positions of the Oval Track puzzle drawn below.



(a) (b)

14. **Oval Track Puzzle move sequence in cycle notation.** Express the move sequence α given in the diagram below as a permutation in cycle notation.



15. For each of the following move sequences, which are applied to the solved-state Oval Track puzzle, draw the resulting configuration of the disks on the puzzle.

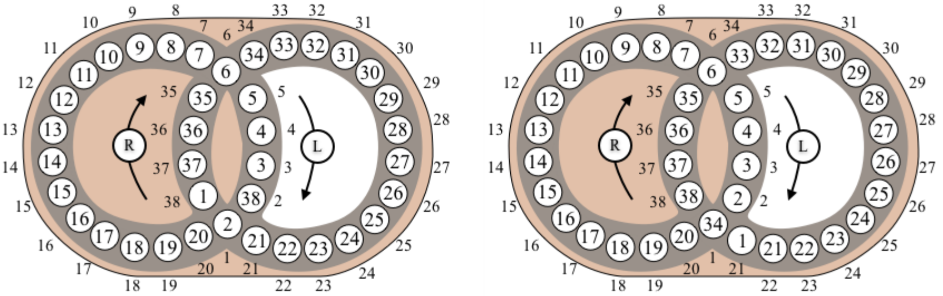
(a) T^2

(b) R^{19}

(c) $R^{-1}TR$

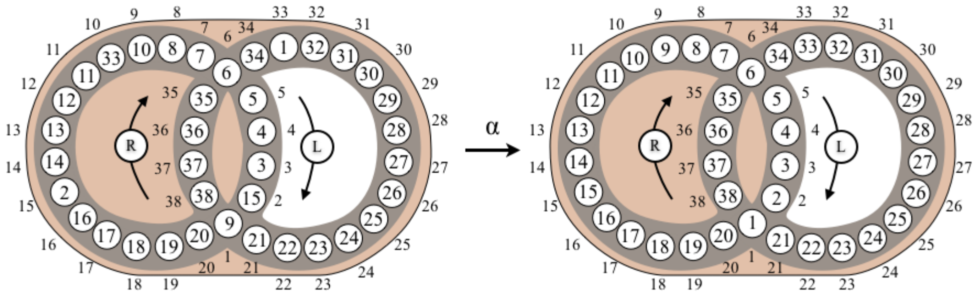
(d) $TR^{-1}TR$

16. **Hungarian Rings arrangements into cycle notation.** Express, in cycle form, the permutation describing each of the positions of the Hungarian Rings puzzle drawn below.



(a) (b)

17. **Hungarian Rings move sequence in cycle notation.** Express the move sequence α given in the diagram below as a permutation in cycle notation.



18. For each of the following move sequences, which are applied to the solved-state Hungarian Rings puzzle, draw the resulting configuration of the disks on the puzzle.

    (a) R^2

    (b) RL

    (c) $L^5R^5L^{-5}R^{-6}LR^6L^5R^{-5}L^{-5}R^{-1}L^{-1}R$ (use Sage to compute this)

19. **Rubik's Cube arrangements into cycle notation.** Express, in cycle form, the permutation corresponding to the position of the Rubik's Cube where the cubies have been moved and positioned as follows:

    - the UR cubie is in the bu cubicle (recall this means the U face of the UR cubie is in the B face of the bu cubicle)
    - the UB cubie is in the lu cubicle
    - the UL cubie is in the ur cubicle.

    (Look back at Lecture 1 where the terms "cubie" and "cubicle" are discussed.)

Lecture 6

Permutations: Products of 2-Cycles

To solve a permutation puzzle one must determine how the permutation representing the current position of the pieces can be decomposed into permutations representing the legal moves. It is this "decomposition problem" that will be the focus of our attention in many lectures to come.

In this lecture we will show every permutation can be decomposed as a product of 2-cycles. We will also see how this is connected to the solvability of the Swap puzzle.

It is standard terminology to refer to a 2-cycle as a **transposition** . So the title of this lecture could also be *Permutations: Products of Transpositions*.

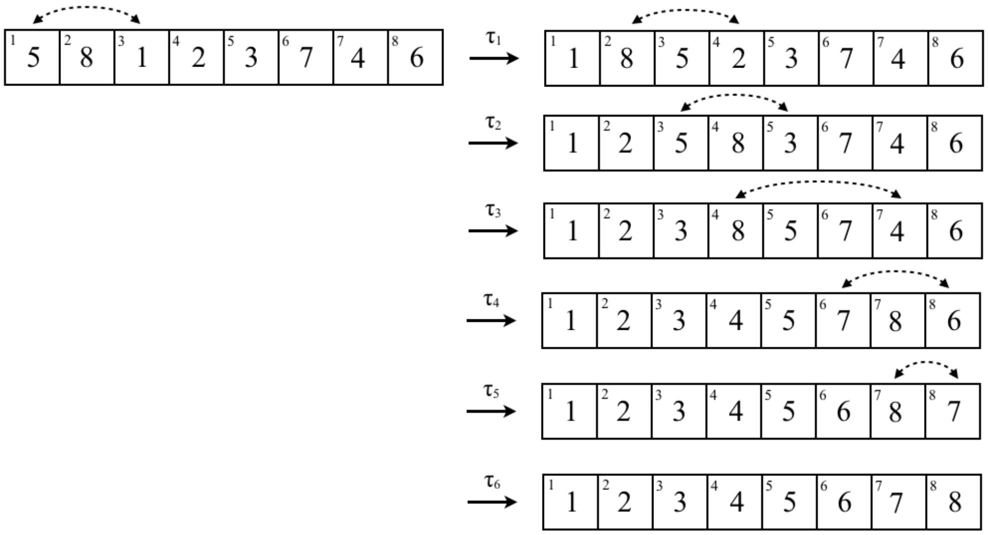
6.1 Introduction

Consider the permutation $\alpha = (1,3,5)(2,4,7,6,8)$. We would like to show it can be written as a product to 2-cycles.

To this permutation we consider the corresponding scramble of the Swap puzzle on 8 objects.

^{1} 5	^{2} 8	^{3} 1	^{4} 2	^{5} 3	^{6} 7	^{7} 4	^{8} 6

To solve the puzzle recall the objective is to restore all numbered tiles to their home positions where the only legal moves are to swap tiles from any two boxes (i.e. a 2-cycle). One possible play is as follows.



The dotted arrows indicate the two tile that are about to be swapped.

The permutations corresponding to the moves are:

$$\tau_1 = (1,3), \quad \tau_2 = (2,4), \quad \tau_3 = (3,5), \quad \tau_4 = (4,7), \quad \tau_5 = (6,8), \quad \tau_6 = (7,8)$$

and so the game-play corresponds to the composition: $\alpha\tau_1\tau_2\tau_3\tau_4\tau_5\tau_6 = \varepsilon$. It follows that

$$\begin{aligned} \alpha &= \tau_6^{-1}\tau_5^{-1}\tau_4^{-1}\tau_3^{-1}\tau_2^{-1}\tau_1^{-1} && \textbf{(6.1)} \\ &= (7,8)(6,8)(4,7)(3,5)(2,4)(1,3) && \textbf{(6.2)} \end{aligned}$$

This is precisely what we wanted, α is written as a product of 2-cycles.

Exercise 6.1: Write the permutation $\beta = (1,5,3,4,2)$ as a product of 2-cycles. Do this by using β as the starting scramble of the Swap puzzle, then solving the puzzle and keeping track of your moves as 2-cycles.

Answer on page 70

6.2 Product of 2-Cycles

There doesn't seem to be anything special about the particular permutation α that we used in the last example. Our strategy was to just move the numbers, one at a time, to their home positions, and we chose to do this in increasing order, though we could have done it an any order we wanted.

This means we should be able to write *any* permutation as a product of 2-cycles. This is such an important observation that will state it as a theorem (a complete proof is given below).

Theorem 6.1 (Product of 2-Cycles): *Every permutation in $S_n, n > 1$, can be expressed as a product of 2-cycles.*

Playing with the Swap puzzle showed us intuitively why the theorem is true, it also gave us a method for finding such a decomposition into 2-cycles. As quick as it was to find a decomposition, we will require a much quicker method: a way to "eyeball" the decomposition. Having to draw a Swap game each time we want to compute a decomposition into 2-cycles would be too time consuming. So how can we do this even more quickly?

Well, consider a 5-cycle: $\beta = (1, 5, 3, 4, 2)$. By direct computation we can check

$$(1, 5, 3, 4, 2) = (1, 5)(1, 3)(1, 4)(1, 2).$$

Check the product for yourself!

In general we have the following "quick" method for decomposing cycles.

Decomposition of a k-cycle into 2-cycles:
A k-cycle $(a_1, a_2, a_3, \ldots, a_{k-1}, a_k)$ in S_n can be decomposed into 2-cycles as follows:

$$(a_1, a_2, a_3, \ldots, a_{k-1}, a_k) = (a_1, a_2)(a_1, a_3)\cdots(a_1, a_{k-1})(a_1, a_k)$$

Using this method of decomposing k-cycles we can easily decompose any permutation by first writing the permutation as a product of disjoint cycles, and then decomposing each cycle into 2-cycles. For example, consider $\alpha = (1, 3, 5)(2, 4, 7, 6, 8)$ again:

$$\alpha = (1, 3, 5)(2, 4, 7, 6, 8) = (1, 3)(1, 5)(2, 4)(2, 7)(2, 6)(2, 8).$$

We now give a formal proof of Theorem 6.1.

Proof: First note that the identity can be expressed as $(1, 2)(1, 2)$, and so it is a product of 2-cycles. (This is why we needed $n > 1$ in the statement of the theorem.) Now consider any permutation $\alpha \in S_n$. We already know we can write α as a product of disjoint cycles:

$$\alpha = (a_1, a_2, \ldots, a_r)(b_1, b_2, \ldots, b_s)\cdots(c_1, c_2, \ldots, c_t)$$

and each cycle can be decomposed into 2-cycles as we observed above:

$$\alpha = (a_1, a_2)(a_1, a_3)\cdots(a_1, a_r)(b_1, b_2)(b_1, b_3)\cdots(b_1, b_s)\cdots(c_1, c_2)(c_1, c_3)\cdots(c_1, c_t).$$

This completes the proof. □

6.3 Solvability of Swap

A permutation α is obtainable as a puzzle position of Swap if and only if it can be expressed as a product of legal moves (2-cycles):

$$\alpha = \tau_k^{-1}\cdots\tau_2^{-1}\tau_1^{-1}.$$

See Equation 6.1 for example. In other words, if α is the current position then the moves required to solve the puzzle are $\tau_1, \tau_2, \ldots, \tau_k$.

Since every permutation is a product of 2-cycles (Theorem 6.1), then as a consequence we have the following:

Corollary 6.2: *The Swap puzzle, where the legal moves consist of swapping contents of any two boxes, is solvable from any configuration. In other words, all permutations in S_n can be obtained in the Swap puzzle on n-objects.*

This is the first in a series of solvability results we wish to obtain for all the puzzles.

Notice, the result only applies to Swap when the legal moves are swapping contents of any two boxes. We could consider other variations of Swap, for example:

Variation 1: Legal moves consist of swapping the contents of any other box with the object in box 1.

For this variation, a permutation α is obtainable as a position if and only if it can be written as a product of 2-cycles of the form: $(1, a)$ for $a \in [n]$. See Exercises 4 and 5.

Variation 2: Legal moves consist of picking any 3 boxes and cycling their contents either to the left or right (i.e. 3-cycles).

For this variation, a permutation α is obtainable as a position if and only if it can be written as a product of 3-cycles. See Exercises 6 and 7.

6.4 Exercises

1. For the permutation $\alpha = (1, 8, 4)(2, 3, 7)(5, 6)$ write it as a product of 2-cycles, first by: (1) Thinking of it as a scrambling of the Swap puzzle, and solving the puzzle as we did in the example in section 6.1, then by (2) Using the method developed in Section 6.2. Which method was the quickest to use?

2. Write the 3-cycle $(1, 2, 3)$ as a product of two 2-cycles.

3. For each of the following permutations, in cycle form, write it as a product of 2-cycles.

   (a) $(1, 6, 4, 3)$

   (b) $(2, 4, 7)(3, 9, 5, 8)$

   (c) $(1, 9, 4, 5)(3, 11, 4)(6, 7)$

   (d) $(1, 2, 3, 4, 5)(6, 7, 8, 9, 10)$

4. Using only the legal moves in Variation 1 of Swap described in Section 6.3, solve the puzzle with initial scrambling $\alpha = (1, 3, 5)(2, 4, 7, 6, 8)$.

5. Show for Variation 1 of Swap described in Section 6.3 that every permutation in S_n is obtainable as a puzzle position. This is equivalent to showing that every permutation in S_n can be written as a product of 2-cycle of the form $(1, a)$ where $a \in [n]$. (Hint: First show every cycle can be written as a product of such transpositions.)

6. Consider only the legal moves in Variation 2 of Swap described in Section 6.3. Determine which of the following scramblings are solvable.
   (a) $\alpha = (1, 3, 5)(2, 4, 7, 6, 8)$
   (b) $\beta = (1, 6, 2)(3, 4, 8)(5, 7)$
   (Hint: Play the game of Swap with these configurations and see if you can solve it.)

7. Discover a solvability condition for Variation 2 of Swap described in Section 6.3. That is, determine the conditions a permutation α must satisfy in order for it to be obtainable as a puzzle configuration. (Given the current tools we have developed so far, this still may be a difficult problem. We'll soon develop the tools needed to completely solve this problem. However, for now see if you can discover a solvability condition.)

Answers to in-chapter exercises:

Exercise 6.1: There are many correct answers to this question. One possible product is $(4, 5)(3, 4)(2, 5)(1, 5)$. Another one is $(1, 5)(1, 3)(1, 4)(1, 2)$.

Lecture 7

Permutations: The Parity Theorem

In this lecture we introduce one of the most important theorems about permutations: **The Parity Theorem**.

We already know every permutation can be expressed using 2-cycles. We now explore the question of how many 2-cycles are needed.

7.1 Introduction

In Lecture 6 we saw that the permutation $\alpha = (1,3,5)(2,4,7,6,8)$ can be written as a product of 2-cycles in two different ways:

$$\begin{aligned}\alpha &= (7,8)(6,8)(4,7)(3,5)(2,4)(1,3)\\ &= (1,3)(1,5)(2,4)(2,7)(2,6)(2,8).\end{aligned}$$

The first decomposition we obtained by considering the permutation as an initial scrambling of the tiles of Swap, then solving the puzzle by restoring each tiles to its home position in increasing order, beginning with tile 1. The second decomposition was obtained using our "quick" method for decomposing permutations. There are many more possible decompositions of α, here are two more:

$$\begin{aligned}\alpha &= (1,6)(6,7)(1,4)(1,7)(2,8)(4,8)(1,5)(3,5)\\ &= (1,3)(1,2)(1,4)(1,2)(1,5)(1,2)(1,7)(1,6)(1,8)(1,2)\end{aligned}$$

The number of 2 cycles used in the decompositions are not always the same. In the four decompositions we have, two use 6 , one uses 8, and one uses 10. Even though the number of 2-cycles isn't constant, it always seems to be of the same parity, in this case it is *even*.

This observation is true in general. Before we state the general result we first better explain the term *parity*. We say for an integer m that its **parity is even** if m is a multiply of 2. If m is not a multiple of 2 then its **parity is odd**. Perhaps the term "parity" is not familiar, but certainly the distinction between an odd number and an even number is. Much in the same way that integers come in one of two types, based on parity: odd or even, permutations also come in one of two types, based on the parity of a permutation. This is what the next theorem says.

Theorem 7.1 (The Parity Theorem): *If a permutation α can be expressed as a product of an even number of 2-cycles, then every decomposition of α into 2-cycles must have and even number. On the other hand, if α can be expressed as a product of an odd number of 2-cycles, then every decomposition of α into 2-cycles must have an odd number. In symbols, if*

$$\alpha = \tau_1\tau_2 \cdots \tau_r = \sigma_1\sigma_2 \cdots \sigma_s$$

where the τ_i's and σ_i's are 2-cycles, then r and s are both even or both odd.

We will give two different proofs of this theorem in the next few sections. But for now lets look at some of the consequences of this theorem.

The Parity Theorem tells us that we can define what we mean by the parity of a permutation. If we decompose it as a product of 2-cycles in any way, then the parity of the number of 2-cycles that we used is either odd or even, and this is the parity we assign to the permutation. Here is the formal definition.

Definition 7.1 (Even and Odd Permutation): A permutation that can be expressed as a product of an even number of 2-cycles is called an **even permutation**. A permutation that can be expressed as a product of an odd number of 2-cycles is called an **odd permutation**.

This definition of parity may not seem to exciting, but just wait. This will allow us to answer important question about the puzzle, and sometimes allow us to abandon quests that are impossible.

Definition 7.2 (Sign of a Permutation): The **sign** of a permutation α is defined to be 1 if α is even, or -1 if α is odd.

$$\text{sign}(\alpha) = \begin{cases} 1 & \text{if } \alpha \text{ is an even permutation,} \\ -1 & \text{if } \alpha \text{ is an odd permutation.} \end{cases}$$

Parity of the Identity:
The identity permutation ε is an even permutation.

This follows from $\varepsilon = (1,2)(1,2)$, which is a decomposition into and even number of 2-cycles.

Sage

```
sage: S5=SymmetricGroup(5)
sage: a=S5("()")   #the identity permutation in cycle form.
sage: a.sign()
1
```

It is useful to be aware of the parity of cycles.

Parity of a Cycle:
An m-cycle, $(a_1, a_2, \ldots, a_m)$ is an even permutation if m is odd, and it is an odd permutation if m is even. (Confusing, I know.)

This follows from the fact that an m-=cycle can be expressed as a product of $m-1$ transpositions:

$$(a_1, a_2, \ldots, a_m) = (a_1, a_2)(a_1, a_3) \cdots (a_1, a_m).$$

If m is even then $m-1$ is odd, and vice-versa. This is why the parity of the permutation is opposite to the parity of the length of the cycle.

Sage

```
sage: S5=SymmetricGroup(5)
sage: a=S6("(1,2,3,4)")
sage: a.sign()
-1
sage: b=S5("(1,2,3,4,5)")
sage: b.sign()
1
```

Example 7.1: Determine whether the following permutations are odd or even.

(a) $(1,5,11,6,7,3)$

(b) $(1,4,12)(3,8,5,9)(7,10)$

(a) This is a 6-cycle and therefore an odd permutation since it can be written as a product of 5 transpositions:

$$(1,5,11,6,7,3) = (1,5)(1,11)(1,6)(1,7)(1,3).$$

Sage

```
sage: S11=SymmetricGroup(11);
sage: a=S11("(1,5,11,6,7,3)");
sage: a.sign()
-1
```

(b) Writing each cycle as a product of transpositions we have:

$$(1,4,12)(3,8,5,9)(7,10) = (1,4)(1,12)(3,8)(3,5)(3,9)(7,10).$$

Since $(1,4,12)(3,8,5,9)(7,10)$ can be written as the product of 6 transpositions, it follows that it is even.

Sage

```
sage: S12=SymmetricGroup(12);
sage: b=S12("(1,4,12)(3,8,5,9)(7,10)");
sage: b.sign()
1
```

7.2 Variation of Swap

in Lecture 6 we considered the following variation on the legal moves of Swap.

Variation: Legal move is to pick any 3 boxes and cycle their contents either to the left or right.

For this variation, a permutation corresponding to a scrambling of the tiles is solvable if and only if it can be expressed as a product of 3-cycles (i.e. the legal moves). Since 3-cycles are even permutations, and products of even permutations are even (see Exercise 5) then any product of 3-cycles must be an even permutation. This means an odd permutation of Swap is not solvable under this variation of the legal moves.

For example, the scrambling

^{1} 2	^{2} 8	^{3} 1	^{4} 5	^{5} 3	^{6} 7	^{7} 4	^{8} 6

is not solvable. This corresponds to the permutation $\alpha = (1,3,5,4,7,6,8,2)$ which is an odd permutation.

Note what happened here. By simply observing that α is an odd permutation we immediately knew to abandon any quest to solve the puzzle. To do otherwise would be pointless. This provides a glimpse into how we will be using the Parity Theorem to investigate the solvability of puzzles.

7.3 Proof of the Parity Theorem

In this section we provide a two different proofs of the Parity Theorem. However, rather than proving the Parity Theorem directly we will prove another result (Claim 7.2), from which the Parity Theorem follows. While reading this section, keep in mind we cannot assume that the Parity Theorem is true (yet), since this is what we are trying to prove.

Consider the following Claim.

Claim 7.1: *Any expression for the identity permutation ε as a product of transpositions uses an even number of them. That is, if*

$$\varepsilon = \tau_1\tau_2\cdots\tau_m$$

where the τ_i's are transpositions, then m is an even integer.

Before considering *why* this is true, let's see how the Parity Theorem is a consequence of this claim. Suppose $\tau_1\tau_2\cdots\tau_r$ and $\sigma_1\sigma_2\cdots\sigma_s$ are two decompositions of a permutation α into 2-cycles. Then

$$\varepsilon = \alpha\alpha^{-1} = (\tau_1\tau_2\cdots\tau_r)(\sigma_1\sigma_2\cdots\sigma_s)^{-1} = \tau_1\tau_2\cdots\tau_r\sigma_s^{-1}\cdots\sigma_2^{-1}\sigma_1^{-1}$$

is a decomposition of ε into $r+s$ transpositions. If Claim 7.1 is true, then $r+s$ must be even, from which it follows that r and s have the same parity. Therefore the Parity Theorem 7.1 is true.

Therefore, in order to prove the Parity Theorem it is sufficient to prove Claim 7.1. But how do we know Claim 7.1 is true? Well, one way is to prove that:

Claim 7.2: *If there is an expression $\tau_1\tau_2\cdots\tau_m$ for the identity permutation ε that uses m transpositions, then there is an expression for ε that uses $m-2$ transpositions.*

Again, before considering *why* Claim 7.2 is true, let's see how we can use it to prove Claim 7.1. Let's assume to the contrary that it was possible to have an expression $\tau_1\tau_2\cdots\tau_m$ for ε where m is odd. Then, assuming Claim 7.2 is true, we could get an expression using $m-2$ transpositions (which is still an odd number of transpositions). We could keep applying Claim 7.2, reducing the number of transpositions by 2 each time, until we end up with an expression for ε using only one transposition. But this is impossible since a single transposition is not equal to the identity (the two numbers in the cycle would not be fixed by the permutation). The fact that we get something impossible from the assumption that an expression for ε exists that uses an odd number of transpositions forces us to conclude that Claim 7.1 is true.

To summarize we have

$$\text{Claim 7.2} \Rightarrow \text{Claim 7.1} \Rightarrow \text{Theorem 7.1}$$

So it suffices to prove Claim 7.2. This is the proof will focus on here. We will provide two completely different proofs, one will be algebraic in nature and will involve playing around with the cycle decomposition of permutations (this is the classic proof), the other will be a little more tactile and has a game-like feel to it (this proof is due John O. Kiltinen).

7.3.1 Proof 1 of Claim 7.2

The rough idea of what we will do is: first we will pick the right-most occurrence of any number appearing in decomposition into transpositions. Then we will push this number to the left through the transpositions, while transforming the transpositions at the same time, until we eventually get two transpositions that cancel.

Before giving a formal proof let's look at an example. The product of the following 12 transpositions is the identity. Check for yourself!

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(3,5)(2,5)(2,3)(1,5)(2,6)(2,4) \tag{7.1}$$

We will transform this product to a product of only 10 transpositions, which still represents the identity. Choose a number appearing in any transpositions. We'll choose 3. Find the right-most transposition containing this number. In this case it would be the transposition $(2,3)$ (the ninth one in the list). We now want to push 3 to the left, so in this product we replace $(2,5)(2,3)$ with the equivalent permutation $(3,5)(2,5)$ (Check for yourself that $(2,5)(2,3) = (3,5)(2,5)$.):

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(3,5)\mathbf{(3,5)(2,5)}(1,5)(2,6)(2,4).$$

Now we can replace $(3,5)(3,5)$ with ε:

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(2,5)(1,5)(2,6)(2,4),$$

which is an expression using 2 fewer transpositions than we started with.

Sometime it may take a few more steps, for example if we decided to use 5 instead of 3, we would have proceeded as follows: Find the right-most transposition containing this number in Equation (7.1). In this case it would be the transposition $(1,5)$. We now want to push 5 to the left, so in this product we replace $(2,3)(1,5)$ with the equivalent permutation $(1,5)(2,3)$, since disjoint cycle commute.

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(3,5)(2,5)(1,5)(2,3)(2,6)(2,4)$$

Next replace $(2,5)(1,5)$ with $(1,5)(1,2)$:

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(3,5)(1,5)(1,2)(2,3)(2,6)(2,4),$$

then replace $(3,5)(1,5)$ with $(1,5)(1,3)$

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(1,5)(3,4)(1,5)(1,3)(1,2)(2,3)(2,6)(2,4).$$

Since $(3,4)$ and $(1,5)$ commute, the two $(1,5)$'s would cancel and we get:

$$\varepsilon = (1,2)(1,3)(1,4)(1,6)(3,4)(1,3)(1,2)(2,3)(2,6)(2,4),$$

which is an expression using 2 fewer transpositions than we started with.

With these two examples behind us, we now give the formal proof.

Proof of Claim 7.2:
Choose a number a that appears in the transposition τ_m. Since $(i,j) = (j,i)$ for any transposition (i,j), the product $\tau_{m-1}\tau_m$ can be expressed in one of the following ways as shown on the left:

$$\begin{aligned}
(a,b)(a,b) &= \varepsilon \\
(a,c)(a,b) &= (a,b)(b,c) \\
(c,d)(a,b) &= (a,b)(c,d) \\
(b,c)(a,b) &= (a,c)(c,b)
\end{aligned}$$

If the first case occurs we may delete $\tau_{m-1}\tau_m$ in the original product and obtain a product for ε using $m-2$ transpositions. In the other three cases we replace the form $\tau_{m-1}\tau_m$ with what appears on the right to obtain a new product of m transpositions that is still the identity, but where the right-most occurrence of

a has now moved one 2-cycle to the left. We now repeat the process, where at each stage either we cancel two 2-cycles (and we're done), or we form a new product where a has moved another 2-cycle to the left. This process must terminate with a product of $(m-2)$ transpositions equal to the identity, because otherwise we have a product of m transpositions equal to the identity in which the only occurrence of a is in the left-most 2-cycle, and such a product does not fix a whereas the identity does. □

This completes the proof of Claim 7.2, and therefore the proof of the Parity Theorem too.

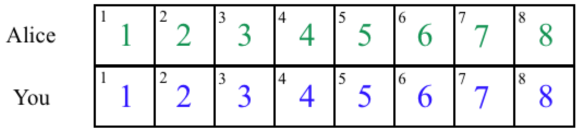
7.3.2 Proof 2 of Claim 7.2

We now present a more tactile proof of Claim 7.2 which is due to John O. Kiltinen (see [9]).

Let's reinterpret what the claim says in terms of the Swap puzzle. Suppose that you start with the Swap puzzle as the identity permutation. Now imagine you do some swaps at random, not paying particular attention to what you are doing. After doing this for a while, you then decide to put everything back in its proper place. In other words you produce a sequence of transpositions that equals ε. If you count the number of total transpositions you used, this number will be even. Try this a few times for yourself. This is precisely what Claim 7.1 says.

Now imagine your friend shows you a sequence of transpositions that they used to produce ε. Is it possible for you to best your friend and produce another such sequence that uses two few transpositions? Claim 7.2 says the answer is yes, and what we'll do here is describe a method to produce the shorter sequence, which can be done in real-time.

Let's call your friend Alice. Imagine two copies of Swap stacked on top of each other, the top is Alice's set and the bottom is yours. We'll colour Alice's tiles green and yours blue, just to make it clear whose is whose.



Alice	1	2	3	4	5	6	7	8
You	1	2	3	4	5	6	7	8

As Alice applies her transpositions, we will match/modify her moves, but in the end we will use two fewer. Here is how we'll do that.

Let's call box containing the tile she touches first the *First Box*. We will call the tile that Alice takes from the First Box *Alice's First Green Tile*, and call the box to which it went the *Tagged Box*. We will call the corresponding blue tile, *Our First Blue Tile*. To aid our memory, let us put markers into these boxes. The marker we place in the First Box (shaded background) will remain throughout the process, but the one in the Tagged Box (solid square in lower right corner) may move, depending on what Alice does.

After Alice makes her first move, we will not make a move of our own. This is the first transposition we omit, the second one is essentially the one she does when she needs to undo this move. Instead of matching her first move, we simply put markers on the boxes. From this point on, however, we will make a move in such a way that the following four conditions are always satisfied, up until she returns Alice's First Green Tile to the First Box, at that point we will just mirror her moves until she finishes.

Conditions to be satisfied after every turn up until she returns Alice's First Green Tile to the First Box:

(i) Alice's First Green Tile is always in the Tagged Box.

(ii) Our First Blue Tile is always at home in the First Box.

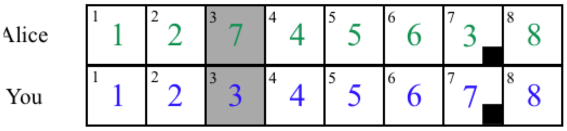
(iii) Whatever green tile number that Alice has in the First Box, we have the blue tile with that number in the Tagged Box.

(iv) All boxes other than the Tagged Box and the First Box contain green tiles and blue tiles with the same numbers.

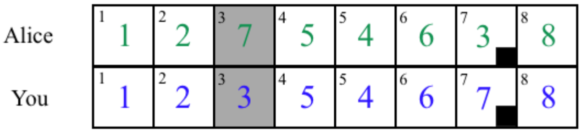
Let's get our hands dirty and do an example. It may look a little confusing at first, but the idea is really straightforward. Suppose Alice's move sequence is

$$(3,7)(4,5)(1,3)(2,7)(3,8)(1,5)(2,6)(3,7)(2,6)(2,3)(1,4)(1,5)(1,7)(1,8).$$

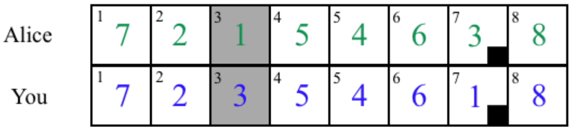
The means Alice's first move is $(3,7)$, and so the First Box is box 3 (shaded background), Alice's First Green Tile is tile 3, and the Tagged Box is box 7 (square box in bottom right corner). So after her first move, we don't make a move, and we have.



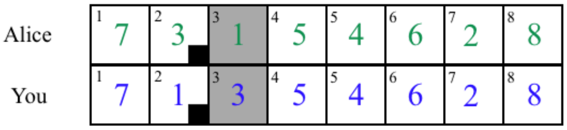
Alice's next move is $(4,5)$. Since she doesn't touch tiles in either the First Box or the Tagged box then conditions (i)-(iv) will remain satisfied if we do the same move: $\tau_1 = (4,5)$.



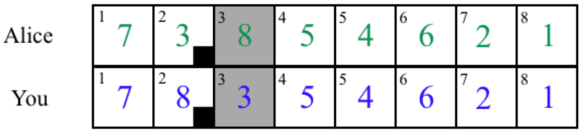
Alice's next move is $(1,3)$, and since this involves the First Box we move the contents of the Tagged Box instead, to satisfy condition (iii): $\tau_2 = (1,7)$.



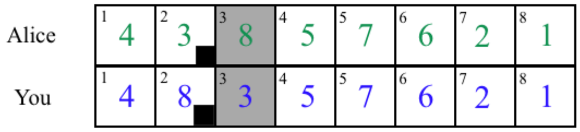
Alice's next move is $(2,7)$, and since this involves the Tagged Box 7 we move the Tagged Box marker as well as the tile, to satisfy condition (i). We also perform the same move on the blue tiles: $\tau_3 = (2,7)$.



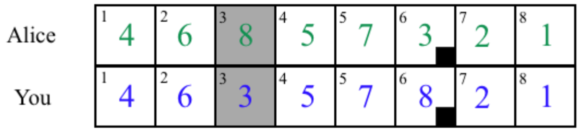
Alice's next move is $(3,8)$, and so we swap the blue tiles in the Tagged Box 2 and box 8. This will keep conditions (i)-(iv) satisfied. $\tau_4 = (2,8)$.



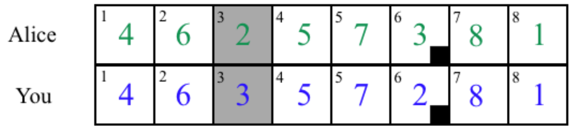
Alice's next move is $(1,5)$ mirror this move: $\tau_5 = (1,5)$.



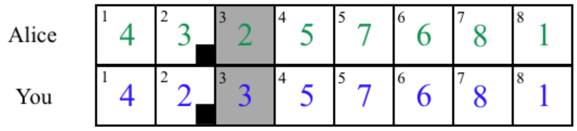
Alice's next move is $(2,6)$ and we mirror it: $\tau_6 = (2,6)$, while at the same time be move the tag to box 6.



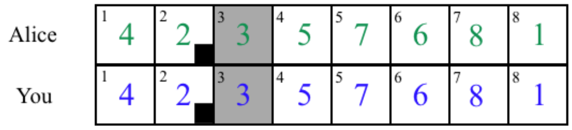
Alice's next move is $(3,7)$ and we do $\tau_7 = (6,7)$.



Alice's next move is $(2,6)$ and we mirror it $\tau_8 = (2,6)$, while at the same time be move the tag to box 2.



Alice's next move is $(2,3)$, and since this involves both the First Box and the Tagged Box this the the other transposition that we skip.



Now both the green and blue tiles are in the same positions so we mirror the remaining moves Alice does: $\tau_9 = (1,4), \tau_{10} = (1,5), \tau_{11} = (1,7), \tau_{12} = (1,8)$. We have now produced as sequence permutations which is the identity:

$$\varepsilon = \prod_{i=1}^{12} \tau_i = (4,5)(1,7)(2,7)(2,8)(1,5)(2,6)(6,7)(2,6)(1,4)(1,5)(1,7)(1,8),$$

but uses 2 fewer permutations than Alice used.

This example illustrates the procedure, but how can we be sure it works in general. That is, how do we know there is always a move that we can do which keeps conditions (i)-(iv) satisfied. Well, the following rules provide us with these moves.

Rules to follow to ensure conditions (i)-(iv) are satisfied:

(a) If Alice does a transposition on her green tiles between boxes other that the First Box or the Tagged Box, then we do the same transposition on our blue tiles.

(b) If Alice does a transposition on her green tiles between the First Box and a box other than the Tagged Box, then we respond with a transposition on our blue tiles between the Tagged Box and the other box, not the First, that she used.

(c) If Alice does a transposition on her green tiles between the Tagged Box and a box other than the First Box, then we respond with a transposition on our blue tiles between the same two boxes. However, we also move the tag so that this other box now becomes the Tagged Box.

(d) If Alice does a transposition on her green tiles between the Tagged Box and the First Box, then we do not do this one.

(e) Once Alice has done a transposition of the type described in (d), which she must, then for every transposition of hers thereafter, we do the same transposition.

If we follow these rules when playing Alice, then we can be certain that conditions (i)-(iv) remain satisfied up until she moves Alice's First Green Tile back to the First Box. She must eventually have to make such a move since Alice's First Green Tile must return home (since the permutation is the identity) and the home position is precisely the First Box. Since we omit the first move and the move where she returns Alice's First Green Tile to the First Box, we have effectively reduced her sequence of transpositions by 2 moves. Thus completing the proof or Claim 7.2.

This may seem like a rather long-winded proof of Claim 7.2, and in fact it is. But this approach is designed to build on the tactile experience that you have developed from playing with the Swap puzzle and other permutation puzzles.

7.4 Exercises

1. Determine whether the following permutations are odd or even.

(a) $(1,3,2)$

(b) $(1,3,5,7,9)$

(c) $(1,6,4,3)$

(d) $(2,4,7)(3,9,5,8)$

(e) $(1,9,4,5)(3,11,4)(6,7)$

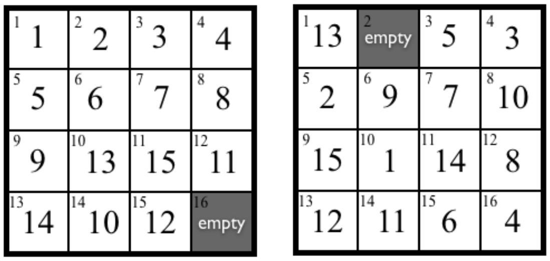
(f) $(1,2,3,4,5)(6,7,8,9,10)$

2. Determine whether the following permutations are odd or even.

(a) $\alpha = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 2 & 7 & 8 & 1 & 4 & 5 & 6 \end{pmatrix}$

(b) $\beta = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 2 & 1 & 6 & 7 & 8 & 3 & 4 \end{pmatrix}$

3. **The parity of 15-puzzle scrambles.** For each of the following arrangements of the 15-puzzle determine the parity of the corresponding permutation.



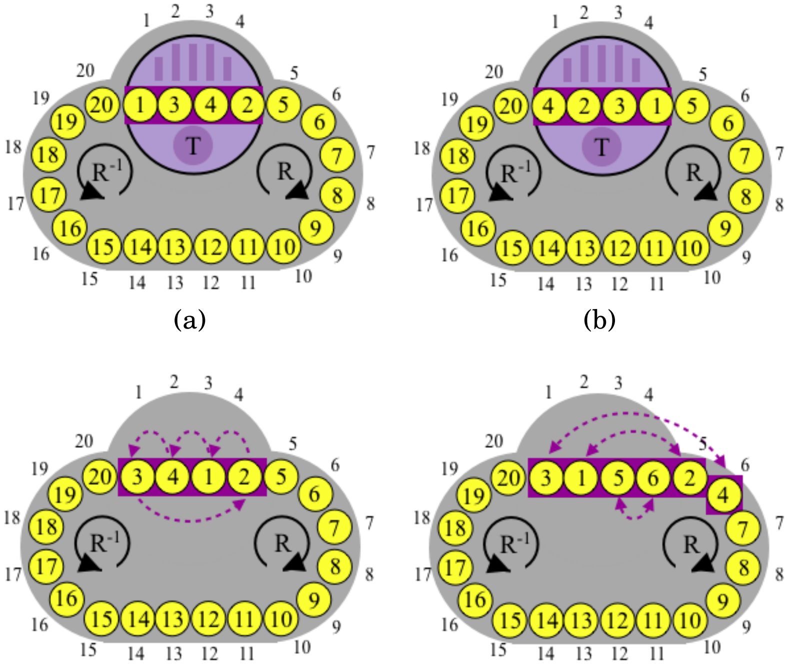
^{1} 1	^{2} 2	^{3} 3	^{4} 4
^{5} 5	^{6} 6	^{7} 7	^{8} 8
^{9} 9	^{10} 13	^{11} 15	^{12} 11
^{13} 14	^{14} 10	^{15} 12	^{16} empty

(a)

^{1} 13	^{2} empty	^{3} 5	^{4} 3
^{5} 2	^{6} 9	^{7} 7	^{8} 10
^{9} 15	^{10} 1	^{11} 14	^{12} 8
^{13} 12	^{14} 11	^{15} 6	^{16} 4

(b)

4. **The parity of some Oval Track end-game scrambles.** For each of the end-game arrangements of the Oval Track puzzle variations, determine its parity.



(a)

(b)

(c)

(d)

5. Show each of the following.

   (a) The product of two even permutations is an even permutation.

   (b) The product of two odd permutations is an even permutation.

   (c) The product of one even permutation and one odd permutation is an odd permutation.

6. In Definition 7.2 we defined the *sign* of an even permutation to be $+1$ and an odd permutation to be -1. Draw and analogy between the result of multiplying two permutations and the result of multiplying their corresponding signs: $+1$ and -1.
(Hint: Use the results of the previous exercise.)

7. If α is even, prove that α^{-1} is even. If α is odd, prove that α^{-1} is odd.

8. Let $\alpha, \beta \in S_n$. Prove that $\alpha^{-1}\beta^{-1}\alpha\beta$ is and even permutation.

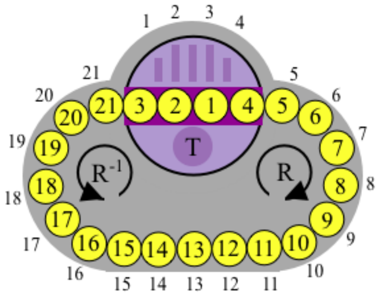
9. Let $\alpha, \beta \in S_n$. Prove that α and $\beta^{-1}\alpha\beta$ have the same parity.

10. Show that exactly half of the permutations in S_n are even.

11. Show that a permutation with odd order must be an even permutation.

12. Give and example of an even permutation with even order. Also give an example of an odd permutation with even order.

13. **Transposition on a Variation of Oval Track.** Consider a variation of the Oval Track puzzle where there are now 21 disks instead of 20. The diagram below shows a configuration in which the tiles in positions 1 and 3 have been swapped. Show it is impossible to solve this configuration.
(Hint: use Exercise 5.)



14. For the $3 \times 3 \times 3$ Rubik's cube, show each of the following.

    (a) It is impossible to find a move sequence that swaps exactly two edge cubies of the Rubik's cube, while leaving every other cubie in its home location.

    (b) It is impossible to find a move sequence that flips exactly one edge cubie of the Rubik's cube, while leaving every other edge cubie in its *home position* (that is, in its home location and with proper orientation).

    (c) It is impossible to find a move sequence that swaps exactly two corner cubies of the Rubik's cube, while leaving every other cubie in its home location.

15. **The least number of transpositions to express a permutation.** Let $\alpha \in S_n$ with disjoint cycle form $\alpha = \sigma_1\sigma_2 \cdots \sigma_r$, where σ_i is a k_i-cycle and they are arranged in such a way that $k_1 \leq k_2 \leq \cdots \leq k_r$. In this situation we say α has cycle structure $(k_1, k_2, \ldots, k_r)$. If we express each cycle as a product of transpositions then we get an expression for α that uses $k - r$ transpositions, where $k = \sum_{i=1}^{r} k_i$. Show that this is the fewest transpositions that there can be in any expression for α in terms of transpositions.
    (See the paper [8] by J.O. Kiltinen for one proof.)
    (Hint: A modification of the ideas used in "Proof 1" of Claim 7.2 may be useful. Note, this gives an *optimal* method for solving the Swap puzzle.)

Lecture 8

Permutations: A_n and 3-Cycles

In this lecture we focus our attention on the set of even permutations, A_n, and show every even permutation can be written as a product of 3-cycles.

8.1 Swap Variation: A Challenge

Consider the following variation of Swap:

Variation: Legal move is to pick any 3 boxes and cycle their contents either to the left or right.

Using only these legal moves, try the following challenges.

Challenge 1: Solve the following puzzle:

^{1} 4	^{2} 8	^{3} 2	^{4} 6	^{5} 5	^{6} 1	^{7} 3	^{8} 7	^{9} 10	^{10} 9

Challenge 2: Solve the following puzzle:

^{1} 1	^{2} 2	^{3} 3	^{4} 4	^{5} 5	^{6} 6	^{7} 7	^{8} 8	^{9} 10	^{10} 9

8.2 The Alternating Group A_n

In Lecture 6 we discovered there are two types of permutations: even and odd. We will denote the **set of all even permutations** by A_n, and the **set of all odd permutations** by O_n. Since every permutation is either odd or even, and no permutation is both, it follows that

$$S_n = A_n \cup O_n, \qquad \text{where } A_n \cap O_n = \emptyset.$$

There is one difference between these two sets which will be important for us, and this has to do with how each of the sets behaves under composition.

> The set A_n of even permutations is closed under composition, closed under taking inverses, and contains the identity. The set O_n of odd permutations is closed under taking inverses, but definitely not closed under composition, nor does it contain the identity. In fact, the composition of an two permutations in O_n is always in A_n.

When we say that A_n (or, in general, *any* subset of B of S_n) is **closed under composition**, we mean that for any $\alpha, \beta \in A_n$ (or in B) the composition $\alpha\beta \in A_n$ (in B). Similarly, by **closed under taking inverses** we mean that for any $\alpha \in A_n$ (or in B) the inverse permutation α^{-1} is also in A_n (in B)

Let's check why our statements about A_n and O_n are true. The product of any two even permutations is another even permutation so A_n is closed under composition. The identity permutation is even and therefore in A_n. For any permutation $\alpha \in A_n$, it's inverse α^{-1} is also even, since once way to express α^{-1} as a product of transpositions is to just write the ones expressing α in reverse order. So if an even number were used to express α then an even number can be used to express α^{-1}. Similarly, if $\beta \in O_n$ then $\beta^{-1} \in O_n$. The product of two odd permutations gives a permutation that can be expressed in terms of an *odd* + *odd* = *even* number of transpositions, and therefore is an even permutation.

This distinction between A_n and O_n will makes A_n a much more important object to study. Why? Well, to answer this we go back to the properties of S_n.

In Lecture 3, we defined the set of all permutations to be the *Symmetric Group*, S_n. We listed various properties this set has, but most notably it has the following four properties regarding composition:

(a) **Closure.** The product of two elements $\alpha, \beta \in S_n$ is another element $\alpha\beta \in S_n$.[1]

(b) **Associativity.** Permutation composition is associative: $\alpha(\beta\gamma) = (\alpha\beta)\gamma$.

(c) **Identity.** The *identity* (or "do nothing") permutation ε is in S_n. It has the property that $\varepsilon\alpha = \alpha\varepsilon = \alpha$ for all $\alpha \in S_n$.

(d) **Inverses.** Every $\alpha \in S_n$ has an *inverse* in S_n denoted by α^{-1}. The defining property of an inverse is $\alpha\alpha^{-1} = \alpha^{-1}\alpha = \varepsilon$.

If we look back at all the computations we've done with permutations we see that we are making extensive use of these properties, whether we are conscious of it or not. For example, the cancellation property: $\alpha\beta = \alpha\gamma$ implies $\beta = \gamma$, and $\beta\alpha = \gamma\alpha$ implies $\beta = \gamma$, is a direct consequence of these four properties. Look back at the proof of it in Lecture 3. This means that any set of objects, equipped with an operation that combines two to produce a third, and the operation satisfies these four properties, also has the cancellation property. For example, $\mathbb{R}$ under the operation of addition, +, satisfies these four properties (identity is 0), so it must also have the cancellation property. The set of invertible 2×2 matrices, under matrix multiplication, satisfies these four properties, so it must also have the cancellation property. In a sense, we have described the "important" properties of S_n.

A set A that comes equipped with an operation to combine pairs of elements (add/multiply/compose) such that A is *closed* under the operation, the operation is *associative*, there is an *identity* in A, and *inverses* exist in A, is called a **group**. Our explorations into permutations puzzles will essentially consist of considering the set of all legal move sequences, call this set M, and noticing that this set is a subset of S_n which is also a group. (Composition of legal moves is a legal move, composition is associative, there is a "do-nothing" move, and for each move there is an way to "undo" it.) Therefore to each permutation puzzle we can associate a group M of legal move sequences. The question is then: Are we able to understand the group M? In order to do this, we'll need to build up our stock of examples of groups.

What we've shown above is that A_n is a group, whereas O_n is not. O_n fails to contain the identity, nor is it closed under composition. A_n is an important family of groups, and in particular A_5 has great historical significance. The letter "A" in its name comes from the word "alternating", which reflects some properties that were important when these groups were first studied.

[1] Our convention is to compose permutations from left-to-right,

Definition 8.1 (Alternating Group of Degree n): The set of even permutations of S_n is denoted by A_n, and is called the **alternating group of degree n**:

$$A_n = \{\alpha \in S_n : \alpha \text{ is an even permutation}\}$$

We will sometimes refer to A_n as the *set of even permutations*. As a first step in investigating A_n, lets show it contains exactly half the elements of S_n.

Theorem 8.1 (Cardinality of A_n): $|A_n| = |O_n| = \dfrac{n!}{2}$, *for* $n \geq 2$.

Proof: To see this we will pair up all the even permutations α with odd permutations $(1,2)\alpha$, to observe there are equal numbers of each.

Consider the set of all elements in S_n of the form $(1,2)\alpha$ where $\alpha \in A_n$, and denote this set by $(1,2)A_n$:

$$(1,2)A_n = \{(1,2)\alpha : \alpha \in A_n\}$$

Observe that $(1,2)A_n \subset O_n$, since extending an even permutation by a transposition is an odd permutation. On the other hand, for $\beta \in O_n$ we have $(1,2)\beta \in A_n$ and so $\beta = (1,2)(1,2)\beta \in (1,2)A_n$. Since β was just any element of O_n this means, $O_n \subset (1,2)A_n$. It follows that $O_n = (1,2)A_n$.

Next we note that $(1,2)A_n$ and A_n have exactly the same number of elements. To see this, we just observe that the function $\phi : A_n \to (1,2)A_n$ defined by $\phi(\alpha) = (1,2)\alpha$ is a bijection. (See Exercise 12.)

Therefore $|A_n| = |O_n|$ and $|A_n| + |O_n| = |S_n|$. Since $|S_n| = n!$ it follows that $A_n = O_n = \frac{n!}{2}$. □

Example 8.1: List the elements of A_4.

These are the permutations in S_4 which are even. The most straightforward way to list elements is to do it in disjoint cycle form, so we'll begin with the identity:

$$\varepsilon.$$

Next, we list elements involving cycles of length at most 2. And since we want even permutations we don't included single transpositions:

$$(1,2)(3,4),\ (1,3)(2,4),\ (1,4)(2,3).$$

Next we can list 3-cycles:

$$(1,2,3),\ (1,3,2),\ (1,2,4),\ (1,4,2),\ (1,3,4),\ (1,4,3),\ (2,3,4),\ (2,4,3).$$

This is all the elements of A_4, and there are 12 as predicted by Theorem 8.1.

8.3 Products of 3-cycles

The fact that every permutation in S_n can be expresses as a product of 2-cycles, is something we have used quite a bit. There is a similar result for the even permutations A_n and 3-cycles.

Theorem 8.2: *Every permutation in A_n, for $n \geq 3$, can be expressed as a product of 3 cycles.*

Proof: Suppose α is an even permutation, then we can express it as the product of an even number of 2-cycles:

$$\alpha = \tau_1\tau_2\cdots\tau_{2k-1}\tau_{2k}.$$

We'll group together adjacent pairs of 2-cycles as follows:

$$\alpha = (\tau_1\tau_2)(\tau_3\tau_4)\cdots(\tau_{2k-1}\tau_{2k}).$$

It suffices to show that a product of two transpositions can either be dropped from the expression or be expressed as a product of 3-cycles, without changing the the value of the expression.

Each product $\tau_i\tau_{i+1}$ can be expressed in one of the following ways as shown on the left, depending on whether the transpositions move two things in common, one thing in common, of nothing in common:

$$\begin{aligned}(a,b)(a,b) &= \varepsilon\\ (a,b)(a,c) &= (a,b,c)\\ (a,b)(c,d) &= (a,b,c)(a,d,c)\end{aligned}$$

If the first case occurs we may delete $\tau_i\tau_{i+1}$ in the original product. In the other two cases we replace $\tau_i\tau_{i+1}$ with what appears on the right to obtain a new product of 3-cycles. □

Example 8.2: Express the even permutation $\alpha = (1,6,4)(2,3,7,8)(9,10)$ as a product of 3-cycles.

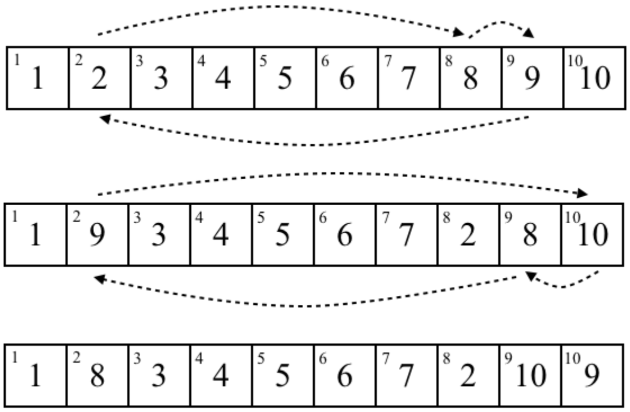
To do this the first thing we do is express it as a product of transpositions:

$$\alpha = (1,6)(1,4)(2,3)(2,7)(2,8)(9,10)$$

Then we group adjacent transpositions and express each in terms of 3-cycles.

$$\begin{aligned}(1,6)(1,4) &= (1,6,4)\\ (2,3)(2,7) &= (2,3,7)\\ (2,8)(9,10) &= (2,8,9)(2,10,9)\end{aligned}$$

It may seem mysterious how we obtained the last one. The following simple game of Swap shows how we can express $(2,8)(9,10)$ as the product of two 3-cycles.



In general, this is precisely the result we used in the proof of the theorem. The way we came up with it there was to look at a simple game of Swap on four objects $a|b|c|d$. To swap a with b and c with d we can first cycle abc to the right: $c|a|b|d$. Then we can cycle objects in positions acd to the left: $b|a|d|c$.

Now we can put everything back together to get:

$$\alpha = (1,6,4)(2,3,7)(2,8,9)(2,10,9).$$

8.4 Variations of Swap: Revisited

Let's go back to the variation of Swap in Section 8.1.

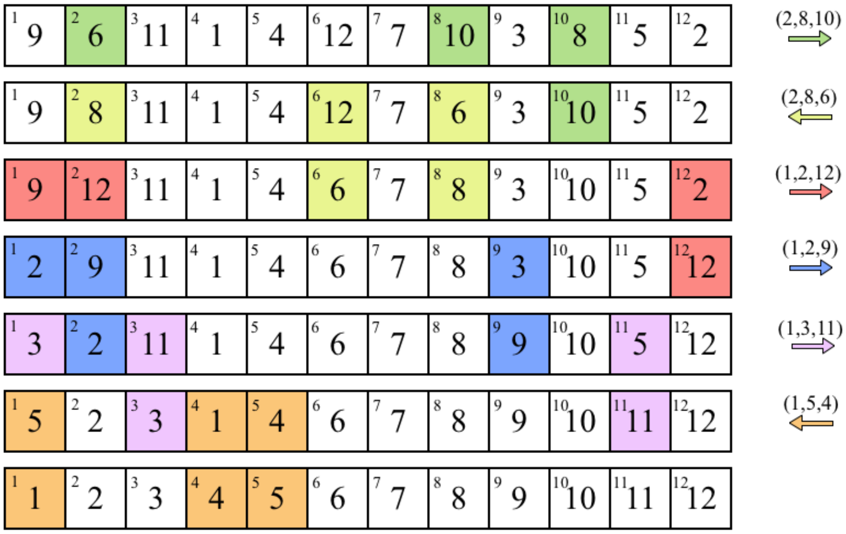
Variation: Legal move is to pick any 3 boxes and cycle their contents either to the left or right.

For example, suppose the puzzle started in the following position:

1	2	3	4	5	6	7	8	9	10	11	12
9	6	11	1	4	12	7	10	3	8	5	2

The corresponding permutation is $\alpha = (1,4,5,11,3,9)(2,12,6)(8,10)$.

We can solve the puzzle as follows. In each line the shaded boxes represent our choice of 3 boxes, and the arrow on the right indicates which direction the contents are being moved. We also summarize the move by writing the corresponding 3-cycle above the arrow.



1	2	3	4	5	6	7	8	9	10	11	12	
9	6	11	1	4	12	7	10	3	8	5	2	$\xrightarrow{(2,8,10)}$
9	8	11	1	4	12	7	6	3	10	5	2	$\xleftarrow{(2,8,6)}$
9	12	11	1	4	6	7	8	3	10	5	2	$\xrightarrow{(1,2,12)}$
2	9	11	1	4	6	7	8	3	10	5	12	$\xrightarrow{(1,2,9)}$
3	2	11	1	4	6	7	8	9	10	5	12	$\xrightarrow{(1,3,11)}$
5	2	3	1	4	6	7	8	9	10	11	12	$\xleftarrow{(1,5,4)}$
1	2	3	4	5	6	7	8	9	10	11	12	

In term of permutations this move sequence tells us:

$$\alpha(2,8,10)(2,8,6)(1,2,12)(1,2,9)(1,3,11)(1,5,4) = \varepsilon$$

or in other words,

$$\begin{aligned}\alpha &= [(2,8,10)(2,8,6)(1,2,12)(1,2,9)(1,3,11)(1,5,4)]^{-1}\\ &= (1,4,5)(1,11,3)(1,9,2)(1,12,2)(2,6,8)(2,10,8).\end{aligned}$$

That is, considering α as a starting position for this variation of Swap, solving the puzzle is equivalent to expressing α as a product of 3-cycles. Since we know only even permutations are expressible as products of 3-cycles this give us a very simple solvability condition for this variation of Swap.

Corollary 8.3 (Solvability of Swap Variation): *The Swap puzzle, where the legal moves consist of 3-cycles on any three boxes, is solvable only when the starting position is an even permutation. In other words, only even permutations can be obtained in this variation of Swap.*

To see this solvability condition in action, consider the following scramble of Swap.

^{1} 2	^{2} 6	^{3} 4	^{4} 1	^{5} 3	^{6} 5

Try solving it using only 3-cycles.

You would very quickly realize it is a difficult task. It is possible to get all but two numbers back into their home positions. In fact, this position corresponds to the permutation $(1, 4, 3, 5, 6, 2)$ which is a 6-cycle, and therefore an odd permutation. Therefore, by Theorem 8.3 no matter how long we play with the puzzle it we don't have a hope of solving it. It is simply impossible!

Looking back at Section 8.1 we see that the permutation in Challenge 1 is $(1, 6, 4)(2, 3, 7, 8)(9, 10)$ which is even and therefore solvable, whereas the permutation in Challenge 2 is $(9, 10)$ which is odd, and therefore not solvable. Just knowing Challenge 1 is solvable doesn't actually answer the question, we were asked to solve the puzzle. This is equivalent to expressing $(1, 6, 4)(2, 3, 7, 8)(9, 10)$ as a product of 3-cycles, which we've already done in Example 8.2. There we found $(1, 6, 4)(2, 3, 7, 8)(9, 10) = (1, 6, 4)(2, 3, 7)(2, 8, 9)(2, 10, 9)$. So applying the inverse of this permutation: $(2, 9, 10)(2, 9, 8)(2, 7, 3)(1, 4, 6)$ will solve the puzzle. On the other hand, knowing the puzzle in Challenge 2 is not solvable means we can abandon playing with it, since it is impossible to solve.

8.5 Exercises

1. Given an example of an element in A_7 which contains a 4-cycle. Give an example of an element in A_{10} which contains at least one 3-cycle, and at least one 4-cycle.

2. Demonstrate the truth of Theorem 8.2 by expressing these even permutations as products of 3-cycles.

   (a) $\alpha = (1, 2)(1, 3)$

   (b) $\beta = (1, 2)(3, 4)$

   (c) $\gamma = (1, 2, 3, 4, 5, 6)(3, 4, 5)(2, 5)(1, 4)(5, 2)$

   (d) $\delta = (1, 2)(2, 3)(4, 5)(1, 3)(6, 7)(6, 8)(9, 10)(11, 12)$

   (e) $\sigma = (1, 2, 3, 4)(2, 3, 4, 5)(4, 5, 6, 7)(8, 9)$

3. **Expressing odd permutations in terms of 3-cycles and one transposition.**
   (a) Show that all odd permutations in S_n can be expressed using exactly one transposition together with zero or more 3-cycles.
   (b) Demonstrate the truth of this claim by expressing these odd permutations with a single transposition and 3-cycles.

   (i) $\alpha = (1, 2, 3, 4, 5, 6)$

   (ii) $\beta = (1, 2, 3, 4)(5, 6, 7)(8, 9, 10)$

   (iii) $\gamma = (2, 5, 3, 7, 6)(3, 5, 8, 4)(6, 8, 2, 1, 9)$

4. Using the solvability condition for the variation of Swap we considered in this section (Corollary 8.3), determine whether each of the following scrambles are solvable. For the ones that are solvable, find a sequence of moves that solve the puzzle.

   (a)

^{1} 3	^{2} 5	^{3} 1	^{4} 6	^{5} 4	^{6} 2

   (b)

^{1} 6	^{2} 9	^{3} 5	^{4} 2	^{5} 1	^{6} 8	^{7} 10	^{8} 3	^{9} 4	^{10} 7

(c)

1	2	3	4	5	6	7	8	9	10
2	9	1	10	6	7	3	5	8	4

(d)

1	2	3	4	5	6	7	8	9	10	11	12
11	10	12	7	2	8	1	5	3	6	4	9

5. What are the possible orders for permutations in A_6? What about A_7?

6. Show that A_5 contains no element of order 15.

7. What is the maximum order of any element in A_{10}?

8. Compute the order of each permutation in A_4. What arithmetic relationship do these orders have with he cardinality of A_4.

9. How many elements of order 5 are there in A_6.

10. How many elements of order 4 are there in A_6.

11. Show that A_5 has 24 elements of order 5, 20 elements of order 3, and 15 elements of order 2.

12. Show that the function $\phi : A_n \to (1,2)A_n$ defined by $\phi(\alpha) = (1,2)\alpha$ is a bijection. (This result is used in the proof of Theorem 8.1.)

13. Let $B \subset S_n$ be a set of permutations such that

    - B is closed under multiplication (i.e. if $\alpha, \beta \in B$ then so is $\alpha\beta$), and
    - $A_n \subset B$, and
    - B contains an odd permutation γ.

    Show that $B = S_n$. In words this says that a set of permutations B that is closed under multiplication and contains every even permutation and at least one odd permutation must contain *every* permutation.

14. **Products of 4-cycles? 5-cycles?** All permutations in S_n are expressible using transpositions, and all permutations in A_n are expressible using 3-cycles, provided $n \geq 3$. Stating this another way, this says that you get all permutations by taking all possible products of 2-cycles, and similarly you get all the even permutations by taking all possible products of 3-cycles. What do you get when you take all possible products of 4-cycles? Or 5-cycles? Or k-cycles? Explore this question and see what you can discover. Note of course that we must assume $n \geq k$ before we can talk about k-cycles in S_n.

Lecture 9

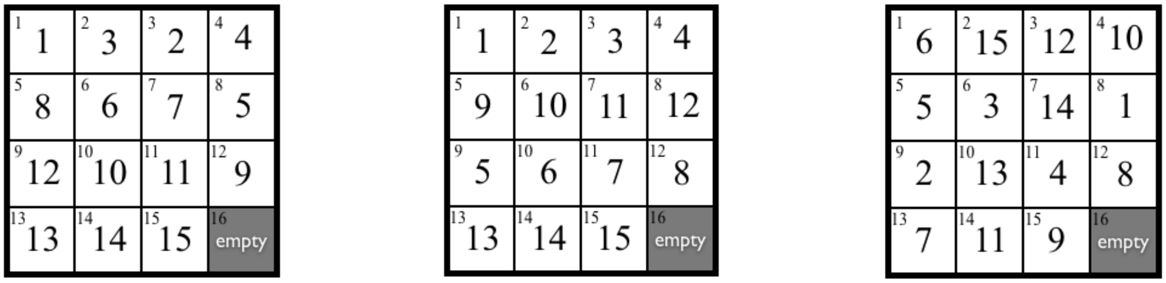
Mastering the 15-Puzzle

We now have enough theory developed to give a full analysis of the 15-puzzle. We will present a solvability criteria which will allow us to easily see whether a given scrambling of the puzzle is solvable. We will also sketch a strategy for solving the puzzle.

9.1 Solvability Criteria

Determining the solvability of a scrambling of the tiles on the 15-puzzle is a simple task as we will see. Let's first consider the case where a scrambling places the empty space back into its original box (box 16). This means the corresponding permutation α fixes 16: $\alpha(16) = 16$. We can think of such a permutation as an element of S_{15}. (Just think about the disjoint cycle form, 16 doesn't appear since it is mapped back to itself.)

Figure 9.1 shows three different configurations of the 15-puzzle corresponding to permutations in S_{15}. The permutations are written below each puzzle. We'd like to be able to quickly determine which configurations are solvable.



(2,3)(5,8)(6,7)(9,12)(10,11)

(a)

(5,9)(6,10)(7,11)(8,12)

(b)

(1,8,12,3,6)(2,9,15)(4,11,14,7,13,10)

(c)

Figure 9.1: Which of the positions are solvable?

The next theorem says any rearrangement of tiles in the 15-puzzle starting from the solved-state configuration which brings the empty space back to its original position must be an even permutation of the other 15 pieces. Moreover, it says that *every* even permutation of the 15 tiles can be obtained as a position on the 15-puzzle.

Theorem 9.1 (Solvability Criteria for 15-Puzzle - Part 1): *A permutation α of the 15-puzzle which fixes* 16*, is solvable if and only if it is even: i.e.* $\alpha \in A_{15}$.

It follows that the number of solvable positions of the 15-Puzzle, where the empty space is in its home position, is

$$|A_{15}| = \frac{15!}{2} = 653,837,184,000.$$

We immediately conclude from this theorem that the puzzles in Figures 9.1a and 9.1c are not solvable since the permutations are odd, whereas the puzzle in Figure 9.1b is solvable since the permutation is even.

We will provide a proof of this theorem in Section 9.2.

What about the case when the scrambling does not place the empty space back in box 16? We'll see that simply knowing the parity of the permutation and the position of the empty space is enough to determine solvability. But first it will be handy to talk about the *parity of a box*.

Colour the 15-puzzle like a checker board as in Figure 9.2. We will call the shaded boxes *even* and the white boxes *odd*, since if the empty space is in a shaded box it takes an even number of moves to bring it to box 16, similarly if it is in a white box it takes an odd number of moves. Under this definition boxes $1, 3, 6, 8, 9, 11, 14, 16$ are even, whereas the other boxes are odd.

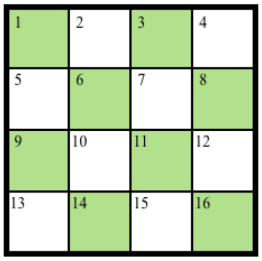


Figure 9.2: **Parity of box:** Define the shaded boxes to be *even* and the white boxes to be *odd*.

With this concept of odd and even boxes now defined, we can state the general solvability criteria for the 15-puzzle.

Theorem 9.2 (Solvability Criteria for 15-Puzzle - Part 2): *A permutation of the 15-puzzle is solvable if and only if the parity of the permutation is the same as the parity of the location of the empty space.*

As a consequence of this theorem when the empty space is in one particular box, there are

$$|A_{15}| = \frac{15!}{2} = 653,837,184,000$$

possible positions of the tiles. Since there are 16 different places to put the empty space, there is a total of

$$16\left(\frac{15!}{2}\right) = \frac{16!}{2} = 10,461,394,944,000$$

possible ways to rearrange the tiles on the board so that the puzzle is solvable. This means, of all $16!$ ways to arrange the tiles in the boxes, exactly half are solvable!

When the puzzle craze hit the world in the early 1880's people noticed that when they randomly placed the tiles in the box, the puzzle was solvable roughly half the time. This now explains why!

As an example, the permutation corresponding to the scrambling in Figure 9.3 is

$$(1, 10, 11, 7, 6)(2, 3, 4, 8, 12, 16, 5)(13, 15)$$

which is odd (check this yourself), and the parity of the location of the empty space is odd, therefore the puzzle is solvable by the solvability criteria: Theorem 9.2.

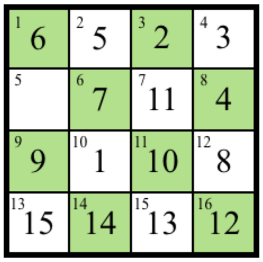


Figure 9.3: Is this position solvable?

9.2 Proof of Solvability Criteria

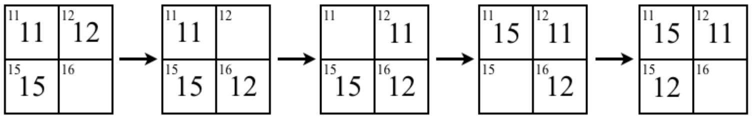
We will prove Theorem 9.1 and then show that Theorem 9.2 is a direct consequence of it.

Proof of Theorem 9.1: There are two directions we need to prove: (i) a solvable configuration is an even permutation, and (ii) every even permutation is a solvable configuration. The proof of (i) is very straightforward, but the proof of (ii) requires us to use the fact that even permutations can be expressed using 3-cycles.

(i) Suppose we have a solvable rearrangement of the 15 tiles, where the empty space is in its home position (box 16). Let $\alpha \in S_{15}$ be the corresponding permutation. Since puzzle moves consist of transpositions - the empty space is swapped with an adjacent tile - then let $\tau_1, \tau_2, \ldots, \tau_k$ be the moves (i.e. transpositions) which solve the puzzle (i.e. takes α to the identity permutation ε). As usual, this means $\alpha = \tau_k \ldots \tau_2\tau_1$. Since the empty space moves around the puzzle and then eventually returns home, the number of moves must be even. To see why this is true, refer to Figure 9.2, the empty space must start in shaded box 16, and after each move it alternates the colour of the box it is in, and so if it returns to a shaded box it must have moved an even number of times. This means k is even, and so α is expressible as a product of an even number of transpositions. Therefore α is even.

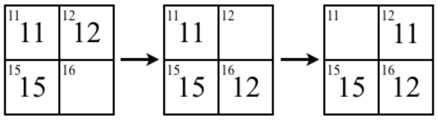
(ii) We wish to show *every* even permutation of the 15 tiles is obtainable through puzzle moves, starting from the solved-state. We will do this by showing we can obtain *any* 3-cycle of the tiles. This is enough to prove the theorem since any even permutation is expressible as a product of 3-cycles, and if we can produce any 3-cycle then we can produce any product of them, through sequential moves, and therefore we can produce any even permutation.

We begin by observing we can produce the 3-cycle $\sigma = (11, 12, 15)$, by focussing on the bottom right corner of the puzzle:

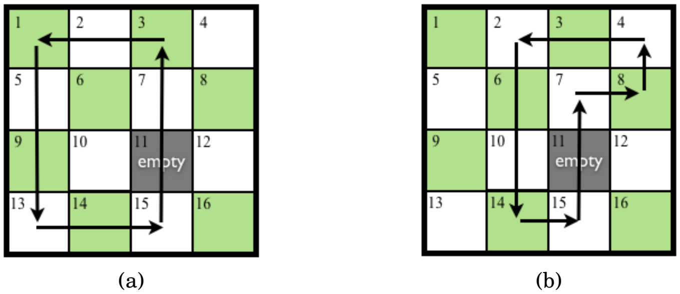


The sequence of moves is: $(12, 16)(11, 12)(11, 15)(15, 16)$

Now that we have one 3-cycle σ, we will show that we can use σ to construct any other 3-cycle we want. From a solved puzzle, pick any tile, say $i \in [15]$. Move tiles 12 and 11 to boxes 16 and 12, respectively, by the move sequence $\alpha = (12, 16)(11, 12)$:



Then using one of the two tours in Figure 9.4 we can move tile i to box 15, without disturbing the contents of boxes 12 and 16. Call this move sequence β.

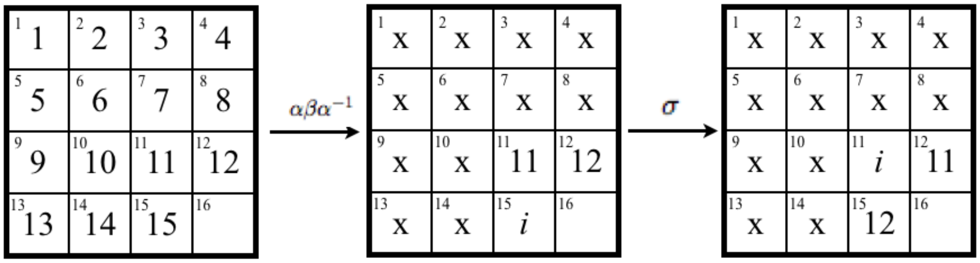


(a)

(b)

Figure 9.4: Tours for producing 3-cycles.

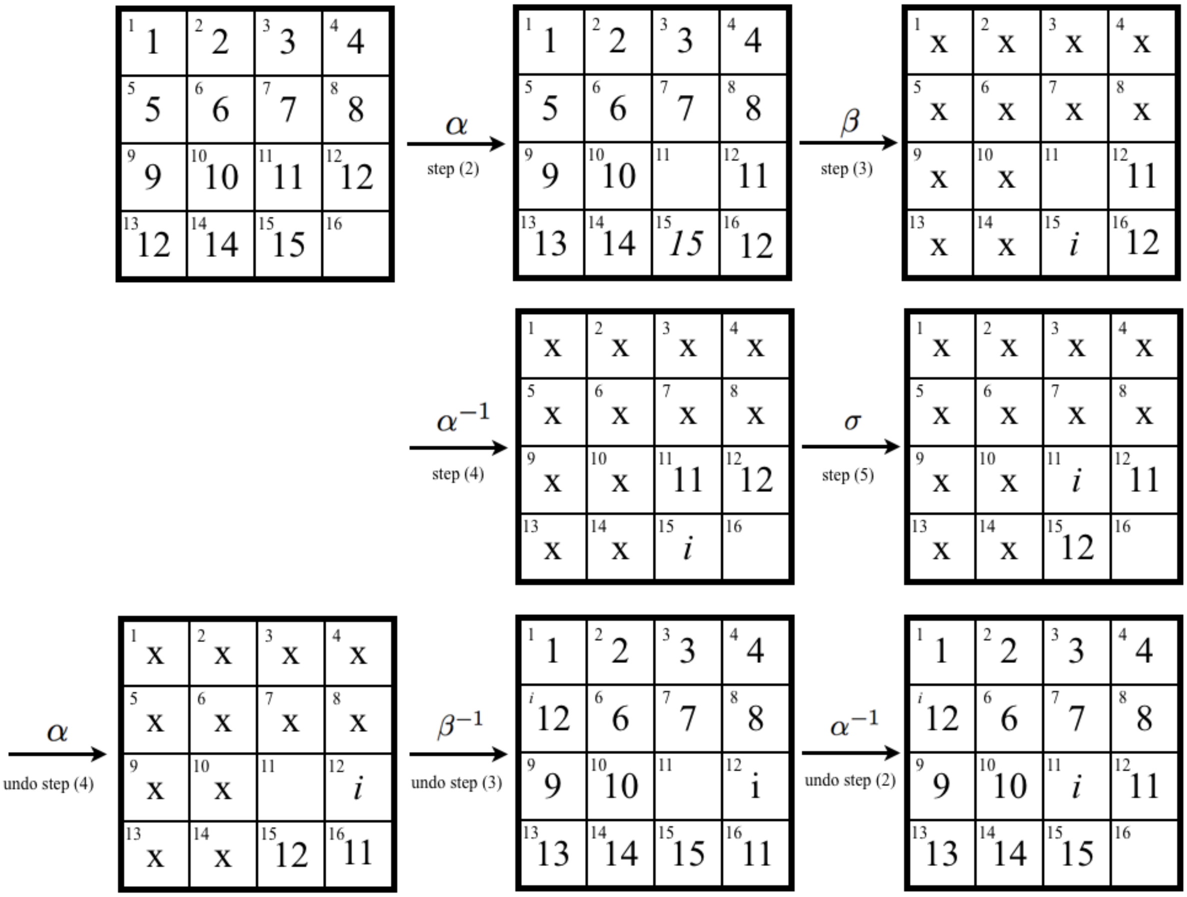
Applying α^{-1} then moves 11 and 12 back into their home positions. This puts the puzzle in the middle position in the following diagram, where the x's indicate these numbers may have moved around.



Applying the 3-cycle σ then moves tiles i, 11, and 12 around as indicated in the diagram. Now, we apply the inverse move sequence $(\alpha\beta\alpha^{-1})^{-1} = \alpha\beta^{-1}\alpha^{-1}$ and this takes everything back to where it was, except i stays in box 11, 11 stays in box 12, and 12 goes to box i. Therefore, we have created the 3-cycle $(11, 12, i)$, where i is any other tile we wish.

Let's summarize what we did:
(1) we chose some tile i,
(2) temporarily hide tiles 11 and 12 in boxes 12 and 16,
(3) used one of the tours in Figure 9.4 to bring tile i to box 15, and this didn't disturb tiles 11 and 12 hidden in boxes 12 and 16
(4) moved 11 and 12 back out to their original positions
(5) applied the 3-cycle $\sigma = (11, 12, 15)$
(6) then reversed all the steps (4) to (2), thus taking everything home except 11,12, and i have been cycled.

Here is an example (near the end we think of i as being 5 for concreteness).



So we now can construct any 3-cycle of the form $(11, 12, i)$, for any $i \neq 11, 12$.

Since $(11, 12, k)(11, 12, j) = (11, j)(12, k)$ we are able to put any tiles (j and k) in boxes 11 and 12, while leaving everything else in place. Moreover,

$$(11, j)(12, k)(11, 12, i)(11, j)(12, k) = (i, j, k),$$

where $i \neq j \neq k$ and $i, j, k \notin \{11, 12\}$. Therefore, we can produce any possible 3-cycle.

This completes the proof. □

The proof of the general solvability condition is a simple consequence of this specific case.

Proof of Theorem 9.2: Let α be the current permutation of the 15-puzzle. Move the empty space to box 16, then the new arrangement corresponds to the permutation

$$\alpha^* = \alpha\tau_1\tau_2 \cdots \tau_k$$

where $\tau_1, \tau_2, \ldots, \tau_k$ were the transpositions used to move the empty space to box 16. Since the empty space is now in box 16 then, by Theorem 9.1, α^* is solvable if and only if it is an even permutation.

Let's think about when α^* is even. This really follows from the way we defined the parities of the boxes.

If the empty space was in an odd box, then it would have taken an odd number of transpositions to move it to box 16. That is, k would be odd. On the other hand, if the empty space was in an even box then k would be even, since it would have taken an even number of transpositions to move it to box 16. In either case, k is equal to the parity of the box the empty space was in. This is precisely the reason we defined the parity of a box as we did.

Now, putting it all together, α is solvable if and only if $\alpha^* = \alpha\tau_1\tau_2\cdots\tau_k$ is even, which is equivalent to α and k having the same parity, which is equivalent to α and the location of the empty space having the same parity. This completes the proof. □

9.3 Strategy for Solution

Of course, in proving Theorem 9.1 we've essentially presented a strategy for solution. First move the empty space into box 16, then the resulting permutation is even, so we may express it as a product of 3-cycles. In practice, our typical method for doing this is first to express it as a product of transpositions, then group pairs of transpositions and express them as a 3-cycle or pair of 3-cycles. We can now use the technique outlined in Section 9.2 to produce each 3-cycle, one-by-one, by moving the desired tiles into the $11, 12, 15$ boxes, performing the 3-cycle $(11, 12, 15)$, then moving everything back again.

Though theoretically possible, and a perfectly sound way to prove the theorem, this makes for a completely inelegant way to solve the puzzle. Not to mention you would need to remember, or write down, the move sequence $\alpha\beta\alpha^{-1}$ since you would need to apply the inverse. This move sequence could be very long. Instead we'll look for a more efficient solution, and one that doesn't require remembering any previously made moves.

Some hints to get you started:

Hint 1: Solve the puzzle by setting the tiles in their proper places, one-by-one, in numerical order. At some stages, it may be necessary to temporarily disturb placed pieces, but they shouldn't have too move to far out of place.

If you haven't tried this already, do so now.

Hint 2: There are some tricky parts. For instance if $1, 2, 3$ are all in place, but 4 is not, it will be necessary to disturb the previously placed pieces in order to get 4 in its proper place. Instead, before placing 3, join it with 4 to form a chain and bring the two of them into place together. Forming chains of tiles is a useful strategy.

If you haven't tried this already, do so now.

Hint 3: Getting the final few pieces in the proper places is of course tricky. But at this stage, making use of 3-cycles, as in the proof above, may be useful. After all, if you can use mathematics to shed some light on what to do, then do it!

Most of all, just have some fun. Try some strategies of your own, if they are useful, then write them down so you won't forget them.

SPOILER ALERT: We now present a complete method for solving the 15-Puzzle, read only if you want to spoil the fun of discovering a solution yourself.

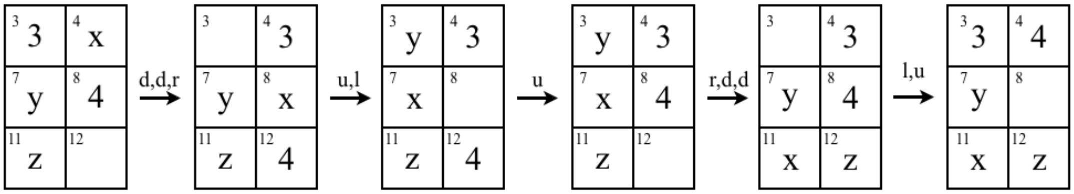
The following solution is due to Jaap Scherphuis (see [13]). It is not an optimal solution, that is, it won't allow you to solve the puzzle in the minimum number of moves, but it does give a method that works on any solvable configuration, and it extends to puzzles of sizes other than 4-by-4. Using this method, with a smooth puzzle, or better yet a virtual version, solutions can take between 1 to 2 minutes, possibly faster.

Phase 1: Solve the top row from left to right.

1. Find the next tile you want to place in position in the top row.
2. If it is not the last tile of the row, it is fairly easy to place correctly, simply keep the following points in mind:
   (a) Never disturb any previously placed pieces.

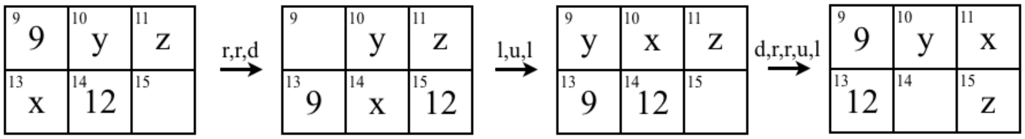
(b) To move the tile in a certain direction, move the other tiles around until the space is next to your tile on the side you want to move it to. Then you can move the tile.

3. If the last tile is not already in position, bring it to the position directly below its correct spot, with the space directly below that. Then move tiles in the following directions: *down, down, right, up, left, up, right, down, down, left, up*. This should place the piece in position. Note it does temporarily disturb the previously placed tile. See figure below.



Phase 2: Solve the rest of the puzzle

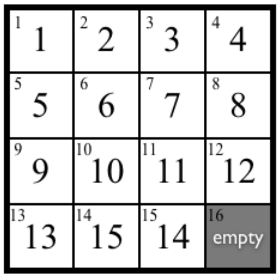
1. Use the technique in phase 1 to solve each row in turn, until there are only two rows left.
2. Rotate the puzzle a quarter turn to the right. The left column of the two rows becomes the top row now.
3. Use the technique in phase 1 to solve each row in turn, until there are only two rows left. This means there is only a 2x2 square left to solve. For example, the next figure show how to get tile 12 in the correct place in the bottom left corner of the 4-by-4 version.



4. Move the pieces in the remaining 2x2 square around until one piece is positioned correctly, and the space is in the correct spot. The other two tiles should automatically be correctly positioned as well.
5. If there are two tiles that need to be swapped, then this cannot be done unless two other tiles are swapped as well. If there are two identical tiles somewhere in the puzzle, then you will have to swap them and solve the rest again. (This may happen if there are letters or pictures on the tiles instead of numbers.)

9.4 Exercises

1. In the early 1880's the world went crazy over trying to solve configuration of the 15-puzzle where the 14 and 15 were swapped. See the Figure below. Explain why no one was able to find a solution.



2. Show that each of the following scramblings of the 15-puzzle are solvable.

^{1} 1	^{2} 5	^{3} 9	^{4} 13
^{5} 2	^{6} 6	^{7} 10	^{8} 14
^{9} 3	^{10} 7	^{11} 11	^{12} 15
^{13} 4	^{14} 8	^{15} 12	^{16} empty

(a)

^{1} 12	^{2} 13	^{3} 14	^{4} 15
^{5} 8	^{6} 9	^{7} 10	^{8} 11
^{9} 4	^{10} 5	^{11} 6	^{12} 7
^{13} 1	^{14} 2	^{15} 3	^{16} empty

(b)

^{1} 4	^{2} 3	^{3} 2	^{4} 1
^{5} 8	^{6} 7	^{7} 6	^{8} 5
^{9} 12	^{10} 11	^{11} 10	^{12} 9
^{13} empty	^{14} 13	^{15} 14	^{16} 15

(c)

^{1} 2	^{2} 4	^{3} 6	^{4} 8
^{5} 10	^{6} 12	^{7} 14	^{8} empty
^{9} 1	^{10} 3	^{11} 5	^{12} 7
^{13} 9	^{14} 11	^{15} 13	^{16} 15

(d)

3. Show that each of the following scramblings of the 15-puzzle are unsolvable.

^{1} 15	^{2} 14	^{3} 13	^{4} 12
^{5} 11	^{6} 10	^{7} 9	^{8} 8
^{9} 7	^{10} 6	^{11} 5	^{12} 4
^{13} 3	^{14} 2	^{15} 1	^{16} empty

(a)

^{1} 1	^{2} 8	^{3} 9	^{4} empty
^{5} 2	^{6} 7	^{7} 10	^{8} 15
^{9} 3	^{10} 6	^{11} 11	^{12} 14
^{13} 4	^{14} 5	^{15} 12	^{16} 13

(b)

^{1} 1	^{2} 2	^{3} 4	^{4} 7
^{5} 3	^{6} 5	^{7} 8	^{8} 11
^{9} 6	^{10} 9	^{11} 12	^{12} 14
^{13} 10	^{14} 13	^{15} 15	^{16} empty

(c)

^{1} 4	^{2} 3	^{3} 2	^{4} 1
^{5} 5	^{6} 14	^{7} 13	^{8} 12
^{9} 6	^{10} 15	^{11} empty	^{12} 11
^{13} 7	^{14} 8	^{15} 9	^{16} 10

(d)

4. Determine which of the following arrangements of the 15-puzzle are solvable and which are unsolvable.

^{1} 1	^{2} 2	^{3} 4	^{4} 3
^{5} empty	^{6} 9	^{7} 11	^{8} 6
^{9} 14	^{10} 13	^{11} 15	^{12} 8
^{13} 5	^{14} 12	^{15} 10	^{16} 7

(a)

^{1} 10	^{2} 9	^{3} 8	^{4} 7
^{5} 11	^{6} empty	^{7} 15	^{8} 6
^{9} 12	^{10} 13	^{11} 14	^{12} 5
^{13} 1	^{14} 2	^{15} 3	^{16} 4

(b)

^{1} empty	^{2} 1	^{3} 2	^{4} 3
^{5} 4	^{6} 5	^{7} 6	^{8} 7
^{9} 8	^{10} 9	^{11} 10	^{12} 11
^{13} 12	^{14} 13	^{15} 14	^{16} 15

(c)

^{1} 1	^{2} 2	^{3} 3	^{4} 4
^{5} 12	^{6} 13	^{7} 14	^{8} 15
^{9} 11	^{10} empty	^{11} 5	^{12} 6
^{13} 10	^{14} 9	^{15} 8	^{16} 7

(d)

5. This exercise is to help us understand the details of the proof in Section 9.2, and to get some practice with creating 3-cycles. Starting with the puzzle in the solved state write down a sequence of moves which will produce each of the 3-cycles:

   (a) $(11, 12, 13)$ **(b)** $(11, 12, 8)$ **(c)** $(11, 8, 13)$.

   Either write the moves using transpositions, or use the words "up", "down", "left", "right", to indicate the direction the tile adjacent to the empty space is moved. It may help to use a physical or virtual version of the puzzle. See the "software" section of the course webpage for links to virtual versions of the puzzle.

6. A 15-puzzle manufacturer wants to sell the puzzle with the tiles already mixed-up, and they want the pattern to be "pretty" so it catches the eye of the customer when sitting on a store shelf. This manufactured version of the puzzle does not allow the pieces to be removed, so the pattern needs to be solvable. They propose to use a pattern where all then even numbered tiles are in the first two rows, and the odd numbered tiles in the last two rows (see the Figure below). They also colour all the even tiles red, so that in the solved state the puzzle will have a pattern of vertical lines. If

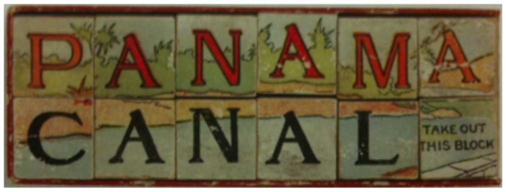
they manufacture the puzzle in this way, will it be solvable? Or will this result in angry customers wanting to return their puzzles?

1 2	2 4	3 6	4 8
5 10	6 12	7 14	8 empty
9 1	10 3	11 5	12 7
13 9	14 11	15 13	16 15

7. In 1959, the Plas-Trix Company in the USA produced a letter version of the 15-puzzle. The problem is to rearrange the blocks so they correctly spell RATE YOUR MIND PAL. They manufactured and sold the puzzle with the last two tiles switched. See the figure below. Explain why it is possible to solve this puzzle.
   (Hint: At first glance it seems this is analogous to the 15-14 problem in Exercise 1, in which case it is not solvable. But this is not entirely equivalent, and the subtle differences are what allows this puzzle to be solved. Can you spot the reason this puzzle is solvable?)

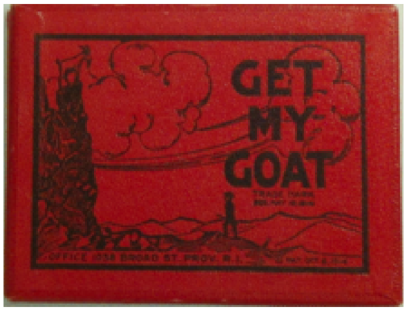
R	A	T	E
Y	O	U	R
M	I	N	D
P	L	A	empty

8. The *Panama Canal Puzzle* dates back to 1915. The staring position has the red letter "P" and the black letter "C" swapped. The problem is to swap the two blocks back. Explain why it is possible to solve this puzzle.



9. The *Get My Goat Puzzle* was patented in 1914. The problem is to get the goat inside the fenced-in area, after removing the marked block. This basically requires a swap of the block with the picture of the goat's head and the block adjacent to it. Explain why this puzzle is solvable.

   You can play an online version of this puzzle at "Nick Baxter's sliding puzzle page" (http://www.johnrausch.com/SlidingBlockPuzzles/).



Conjugation:

Exercises 10 through 12 introduce the idea of *conjugation*.
First a definition:

If $\alpha, \beta \in S_n$, we call the permutation $\beta^{-1}\alpha\beta$ the **conjugate** of α by β.

Looking back at the proof of Theorem 9.1 we transformed the 3-cycle $\sigma = (11, 12, 15)$ into another 3-cycle $(11, 12, i)$ by:

$$\gamma\sigma\gamma^{-1} = (11, 12, i),$$

where $\gamma = \alpha\beta\alpha^{-1}$ was a sequence of moves that moved tile i to box 15, and left tiles 11 and 12 alone. We used conjugation twice in the proof: $\alpha\beta\alpha^{-1}$ and $\gamma\sigma\gamma^{-1}$. These types of products are used extensively when solving permutation puzzles. If you have some experience with permutation puzzles you will notice you frequently make moves of the form:

- do a move m_1,
- then do another move m_2,
- then undo the first move m_1^{-1}.

If you notice you do this, then you already have a working feel for conjugation. In the next few exercises we investigate conjugation, and show that $\beta^{-1}\alpha\beta$ and α have the same cycle structure. This general result is the reason why $\gamma\sigma\gamma$ is a 3-cycle.

10. For each of the following pairs of permutations $\alpha, \beta \in S_n$ calculate the conjugate of α by β. In other words, compute the product $\beta^{-1}\alpha\beta$.

    (a) $\alpha = (1, 2, 3, 4, 5)$, $\beta = (1, 5, 8)(2, 6)(3, 7, 4)$
    (b) $\alpha = (1, 5, 8)(2, 6)(3, 7, 4)$, $\beta = (1, 2, 3, 4, 5)$
    (c) $\alpha = (5, 7, 3, 6)(10, 11, 8, 12)$, $\beta = (1, 2)(4, 10, 5, 11, 7, 9, 12)$

    In each case notice, the cycle structure of $\beta^{-1}\alpha\beta$ is the same as α. For instance in (b), α is a product of two 2-cycles and one 3-cycle, and so is $\beta^{-1}\alpha\beta$.

11. For $\alpha = (1, 2, 3, 4)$ and $\beta = (1, 4)(3, 5, 2)$ do the following.

    (a) Calculate $\beta^{-1}\alpha\beta$.
    (b) Calculate the values of $\beta(1)$, $\beta(2)$, $\beta(3)$, $\beta(4)$, then write down the 4-cycle, $(\beta(1), \beta(2), \beta(3), \beta(4))$.
    (c) Observe that the 4-cycle in part (b) is the same as the answer to part (a). Coincidence? The next exercise says, this is no coincidence.

12. Show that for any $\alpha, \beta \in S_n$ the conjugate $\beta^{-1}\alpha\beta$ has the same cycle structure as α.

    Hint: express α in disjoint cycle form $\sigma_1\sigma_2 \cdots \sigma_k$, where σ_i is a m_i-cycle, for $1 \le i \le k$. Then show

    (i) $\beta^{-1}\alpha\beta = (\beta^{-1}\sigma_1\beta)(\beta^{-1}\sigma_2\beta) \cdots (\beta^{-1}\sigma_k\beta)$.

Then it suffices to only consider the case when α is a cycle. Which means, you just need to prove:

$$\text{(ii)} \quad \beta^{-1}(a_1, a_2, \ldots, a_m)\beta = (\beta(a_1), \beta(a_2), \ldots, \beta(a_m)).$$

Other Board Sizes and Obstacles:

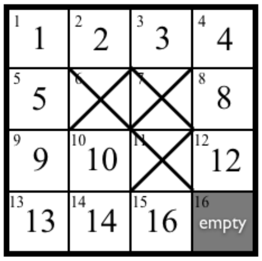
In Exercises 13 to 15 we investigate board sizes other than 4×4.

13. Consider 5 tiles and an empty space on a board consisting of 3 rows and 2 columns. Show, by using a similar argument to the one used for the 15 puzzle that a permutation $\alpha \in S_5$, where the empty space is in its home location, corresponds to a solvable configuration if and only if α is an even permutation.

    Hint: By using a two-by-two square of four boxes, show that a single 3-cycle can be obtained. Then show every 3-cycle can be obtained by conjugation, similar to the argument we used for the 15 puzzle..

14. The following board shows a variation of the 15 puzzle where boxes 6,7, and 11 are obstacles. That is, these boxes are "out-of-play" and cannot be used. We can still ask the question as to which permutations of the tiles are solvable. Show that, just like the original 15 puzzle, Theorems 9.1 and 9.2 remain true.

    Hint: For simplicity just focus on permutations leaving the empty space in its home location. Use the two-by-two square of four boxes to generate all 3-cycles: first show you can obtain one 3-cycle, then use conjugation to obtain all others.



15. **[Challenging]** The following is a general characterization of the solvability condition for rectangular boards, with obstacles. Verify it is true.

    Let α denote an arbitrary permutation of tiles on a rectangular $m \times n$ board such that the board

    (a) has one empty space,

    (b) has at least one two-by-two array of boxes all of which are in use, and

    (c) may have some obstacles (boxes that are out-of-play and cannot be used), but these obstacle do not trap tiles (in other words, any tile can be moved to any other location).

    Then the permutation α is solvable if and only if the parity of α is the same as the parity of the location of the empty space.

Lecture 10

Groups

Group Theory is typically referred to as the mathematical study of symmetry. The puzzles we are studying have exhibited a remarkable amount of symmetry. In this lecture we begin our introduction into group theory by introducing the concept of a *group*. Though, from our experience in exploring puzzles and permutations we already have experience in working with groups. In later lectures, we will see that group theory is the tool required to understand permutation puzzles, in particular their *end-game*.

10.1 Group: Definition

Playing with permutation puzzles has already given us a working definition of a *group*. We have a set of move-sequences, call this set M. We are able to compose two move-sequences together to form a new move-sequence ($m_1, m_2 \in M \implies m_1m_2 \in M$), there is a "do-nothing" move ($\varepsilon \in M$) and we can "undo" a move sequence (for $m_1 \in M$ there is an $m_1^{-1} \in M$). This is also how permutations behave under composition. Each consist of a set, an operation to combine objects in the set, and a few properties this operation must possess. This is precisely what we will call a group.

Definition 10.1 (Group): A **group** is a nonempty set G, together with an operation, which can be thought of as a function $* : G \times G \to G$, that assigns to each ordered pair (a, b) of elements in G an element $a * b \in G$, that satisfies the following properties:

1. *Associativity*: The operation is associative: $(a * b) * c = a * (b * c)$ for all $a, b, c \in G$.
2. *Identity*: There is an element e (called the identity) in G, such that $a * e = e * a = a$ for all $a \in G$.
3. *Inverses*: For each element $a \in G$, there is an element b in G (called the inverse of a) such that $a * b = b * a = e$.

Typically we drop the notation * for the operation and just write the operation by juxtaposition, that is, we simply write $a * b$ as ab. We've already been doing this with permutation composition, and the composition of puzzle moves. In the case were the group operation is addition, then we will use the symbol "+".

Definition 10.2 (Order of a Group): The number of elements of a group (finite or infinite) is called the **order** of the group. We will use $|G|$ to denote the order of the group, since this is really just the cardinality of the set.

The power of mathematics resides in abstraction. Mathematicians look for the similarities between objects, then articulate and abstract these similarities. They generally work with these abstract conceptualizations, since as a result, their discoveries hold for *all* objects satisfying the properties of the abstraction.

Consider an analogy from biology. Biologists consider the similarities between spiders, scorpions, harvestmen, ticks, and mites, to be significant enough that they talk about them as being from the same "family": the Arachnid family. Arachnids are a class of joint-legged invertebrate animals, all of which have eight legs. There are over 100,000 named species, five of which we named above. In this sense, a biologist who studies the (abstract) family Arachnida is in effect studying over 100,000 named species, simultaneously.

Looking back at the definition of a group, in particular at the property "inverses", we see that nowhere did it say the inverse has to be unique. However, in our examples of puzzle movements, and permutations, inverses were unique. Should we have added this as a property? Well, it turns out that we don't need to since it is a direct consequence of the properties in the definition. We'll state this as a theorem.

Theorem 10.1 (Uniqueness of Inverses): *For each element a in a group G, there is a unique element $b \in G$ such that $ab = ba = e$.*

Proof: Suppose b and c are both inverses of a. Then

$$\begin{aligned} b &= be && \text{identity (property 2)} \\ &= b(ac) && \text{since } ac = e \text{ (property 3)} \\ &= (ba)c && \text{by associativity (property 1)} \\ &= ec && \text{since } ba = e \text{ (property 3)} \\ &= c && \text{identity (property 2).} \end{aligned}$$

Therefore $b = c$, so inverses are unique. □

Since inverses are unique we can unambiguously denote the inverse of $a \in G$ by a^{-1}.

Previously we observed that permutations under composition satisfied the cancellation property. This is true of any group.

Theorem 10.2 (Cancellation Property): *In a group G, the right- and left- cancellation properties hold: $ba = ca$ implies $b = c$, and $ab = ac$ implies $b = c$.*

Proof: If $ba = ca$ then $(ba)a^{-1} = (ca)a^{-1}$ and by associativity, $b(aa^{-1}) = c(aa^{-1})$. Since $aa^{-1} = e$, then $be = ce$ from which it follows that $b = c$. Left cancellation can be proved in a similar manner. □

10.1.1 Multiplication (Cayley) Table

Since a group is merely a set with a way to combine elements (a sort-of *multiplication*), we can give the operation in terms of a table, provided the set is finite.

The **multiplication table**[1] of a (finite) group G is a tabulation of the values of the operation $*$. Let $G = \{g_1, ..., g_n\}$. The multiplication table of G is:

$*$	g_1	g_2	$\dots$	g_j	$\dots$	g_n
g_1						
g_2						
$\vdots$						
g_i				$g_i * g_j$		
$\vdots$						
g_n						

This says the entry of the table in row g_i and column g_j is the element $g_i * g_j$.

This table must satisfy some basic properties, which are immediate consequences of the definition of a group:

Lemma 10.3: *(a) Each element $g_k \in G$ occurs exactly once in each row of the table.*
(b) Each element $g_k \in G$ occurs exactly once in each column of the table.
(c) If the $(i, j)^{th}$ entry of the table is equal to the $(j, i)^{th}$ entry then $g_i * g_j = g_j * g_i$.
(d) If the table is symmetric about the diagonal $\searrow$ then $g * h = h * g$ for all $g, h \in G$. (In this case, we call G abelian.)

The proof is left to the reader as Exercise 30.

In the next section we give a number of examples of groups, most of which should already be familiar to the reader. It is interesting to note that we now know, for those examples and any others we encounter during the rest of our lives, if the set satisfies the properties of a group then (i) inverses are unique, and (ii) the cancellation property holds. This is the power of abstraction!

10.2 Some Everyday Examples of Groups

Now that we have a formal description of a group, our first job is to notice we already know many examples.

(1) The set of integers $\mathbb{Z}$, the set of rational numbers $\mathbb{Q}$, and the set of real numbers $\mathbb{R}$, are all groups under ordinary addition. The identity is 0 in each case, and the inverse of a is its negative $-a$.

(2) The set of non-zero rational numbers $\mathbb{Q}^* = \{r \in \mathbb{Q} \mid r \neq 0\}$ is a group under ordinary multiplication. The identity is 1, and the inverse of r is $1/r$.

Similarly, the set of non-zero real numbers $\mathbb{R}^* = \{r \in \mathbb{R} \mid r \neq 0\}$ is a group under ordinary multiplication. The identity is 1, and the inverse of r is $1/r$.

Note, that we had to leave out 0, since it doesn't have a multiplicative inverse, i.e. there is no rational number r such that $r \cdot 0 = 1$. In other words, $\mathbb{Q}$ is not a group under multiplication.

The set of non-zero integers $\mathbb{Z}^* = \{n \in \mathbb{Z} \mid n \neq 0\}$ is *not* a group under ordinary multiplication, since it is not closed under taking inverses. For example, the inverse of 2 is $\frac{1}{2}$, but $\frac{1}{2}$ is not in $\mathbb{Z}^*$.

[1] Also known as a *Cayley table*, after noted English mathematician Arthur Cayley (1821-1895)

(3) The set $\mathbb{R}^3 = \{(a_1, a_2, a_3) \mid a_1, a_2, a_3 \in \mathbb{R}\}$ is a group under componentwise addition:

$$(a_1, a_2, a_3) + (b_1, b_2, b_3) = (a_1 + b_1, a_2 + b_2, a_3 + b_3).$$

The identity is $(0, 0, 0)$ and the inverse of (a_1, a_2, a_3) is $(-a_1, -a_2, -a_3)$.

In general, the set of all n-tuples of real numbers $\mathbb{R}^n = \{(a_1, a_2, \ldots, a_n) \mid a_1, a_2, \ldots, a_n \in \mathbb{R}\}$ is a group under componentwise addition:

$$(a_1, a_2, a_3, \ldots, a_n) + (b_1, b_2, b_3, \ldots, b_n) = (a_1 + b_1, a_2 + b_2, a_3 + b_3, \ldots, a_n + b_n).$$

The identity is $(0, 0, 0, \ldots, 0)$.

(4) A rectangular array of the form $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, where $a, b, c, d \in \mathbb{R}$, is called a 2×2 (real) matrix. The set of all 2×2 matrices is denoted by $M_{2,2}(\mathbb{R})$:

$$M_{2,2}(\mathbb{R}) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mid a, b, c, d \in \mathbb{R} \right\}.$$

If we define the addition of two matrices to be componentwise:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} w & x \\ y & z \end{bmatrix} = \begin{bmatrix} a + w & b + x \\ c + y & d + z \end{bmatrix},$$

then $M_{2,2}(\mathbb{R})$ is a group under this addition. The identity is $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ and the inverse of $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is $\begin{bmatrix} -a & -b \\ -c & -d \end{bmatrix}$.

In general, for positive integers n and m, the set of all matrices with n rows and m columns, the so-called $n \times m$ matrices, $M_{n \times m}(\mathbb{R})$ is a group under componentwise addition.

$$M_{n,m}(\mathbb{R}) = \left\{ \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ a_{n,1} & a_{n,2} & \cdots & a_{n,m} \end{bmatrix} \mid a_{i,j} \in \mathbb{R} \right\}.$$

(5) **General Linear Group.** The *determinant* of a 2×2 matrix $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is the number $\det(A) = ad - bc$. The set of all 2×2 matrices with non-zero determinant,

$$GL(2, \mathbb{R}) = \{A \in M_{2,2}(\mathbb{R}) \mid \det(A) \neq 0\}.$$

under matrix multiplication:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} w & x \\ y & z \end{bmatrix} = \begin{bmatrix} aw + by & ax + bz \\ cw + dy & cx + dz \end{bmatrix}$$

is a group. The identity is $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and the inverse of $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is $\begin{bmatrix} \frac{d}{ad-bc} & \frac{-b}{ad-bc} \\ \frac{-c}{ad-bc} & \frac{a}{ad-bc} \end{bmatrix}$.

In general, the set of invertible $n \times n$ matrices $GL(n, \mathbb{R})$, under matrix multiplication, is a group. It is called the *general linear group of* $n \times n$ *matrices over* $\mathbb{R}$. This follows from the properties that $\det(AB) = \det(A)\det(B)$ and A is invertible if and only if $\det(A) \neq 0$. These statements are proved in any elementary course in linear algebra.

(6) **Special Linear Group.** The set of $n \times n$ matrices with determinant 1 is a group under matrix multiplication. This group is denoted by $SL(n, \mathbb{R})$ and is called the *special linear group of* $n \times n$ *matrices over* $\mathbb{R}$.

$$SL(n, \mathbb{R}) = \{A \in GL(n, \mathbb{R}) \mid \det(A) = 1\}.$$

To see why it is closed under multiplication, suppose $A, B \in SL(n, \mathbb{R})$. Then $\det(A) = 1$ and $\det(B) = 1$, but then $\det(AB) = \det(A)\det(B) = 1 \cdot 1 = 1$, by the property of determinants. Therefore, $AB \in SL(n, \mathbb{R})$. Moreover, since $\det(A^{-1}) = \frac{1}{\det(A)} = \frac{1}{1} = 1$ then $A^{-1} \in SL(n.\mathbb{R})$.

(7) **Differentiable functions.** The set of all differentiable functions $\mathbb{R} \to \mathbb{R}$ is a group under the operation of addition: $(f + g)(x) = f(x) + g(x)$. The reason that the sum of two differentiable functions is differentiable follows from the fact that $\frac{d}{dx}(f + g) = \frac{d}{dx}f + \frac{d}{dx}g$. The reason the (additive) inverse of f is differentiable follows from the fact that $\frac{d}{dx}(-f) = -\frac{d}{dx}f$. The identity is the constant function which maps everything to 0.

(8) **Translations.** For each $(a, b) \in \mathbb{R}^2$, define $T_{a,b} : \mathbb{R}^2 \to \mathbb{R}^2$ by

$$(x, y) \mapsto (x + a, y + b).$$

The set of all such functions $T_{a,b}$:

$$\mathcal{T}(\mathbb{R}^2) = \{T_{a,b} \mid a, b \in \mathbb{R}\}$$

is a group under function composition. To see this, notice that

$$(T_{a,b} \circ T_{c,d})(x, y) = T_{a,b}(T_{c,d}(x, y)) = T_{a,b}(x + c, y + d) = (x + a + c, y + b + d) = T_{a+c,b+d}(x + y)$$

for all $(a, y) \in \mathbb{R}^2$. Therefore, $T_{a,b} \circ T_{c,d} = T_{a+c,b+d}$, so $\mathcal{T}(\mathbb{R}^2)$ is closed under composition. Moreover, $T_{0,0}$ is the identity, and the inverse of $T_{a,b}$ is $T_{-a,-b}$. Function composition is always associative. The elements in $\mathcal{T}(\mathbb{R}^2)$ are called *translations* of $\mathbb{R}^2$.

Similarly we could define the group of translations of $\mathbb{R}^n$, for any positive integer n, as

$$\mathcal{T}(\mathbb{R}^n) = \{T_{a_1,\ldots,a_n} : \mathbb{R}^n \to \mathbb{R}^n \mid a_i \in \mathbb{R}\}$$

where $T_{a_1,\ldots,a_n}(x_1, \ldots, x_n) = (x_1 + a_1, \ldots, x_n + a_n)$.

(9) **Linear Transformations.** A *linear transformation* of $\mathbb{R}^n$ is a function $T : \mathbb{R}^n \to \mathbb{R}^n$ such that $T(a\vec{v} + \vec{w}) = aT(\vec{v}) + T(\vec{w})$ for all $\vec{v}, \vec{w} \in \mathbb{R}^n$ and $a \in \mathbb{R}$. The set of all linear transformations $L(\mathbb{R}^n)$ of $\mathbb{R}^n$, for a positive integer n:

$$L(\mathbb{R}^n) = \{T : \mathbb{R}^n \to \mathbb{R}^n \mid T \text{ is a linear transformation}\}$$

is a group under function addition: for $T, U \in L(\mathbb{R}^n)$ define $T + U$ by

$$(T + U)(\vec{v}) = T(\vec{v}) + U(\vec{v}).$$

To see why, first we note that $T + U$ is a linear transformation since

$$\begin{aligned}(T + U)(a\vec{v} + \vec{w}) &= T(a\vec{v} + \vec{w}) + U(a\vec{v} + \vec{w}) = aT(\vec{v}) + T(\vec{w}) + aU(\vec{v}) + U(\vec{w}) \\ &= a(T(\vec{v}) + U(\vec{v})) + (T(\vec{w} + U(\vec{w})) \\ &= a(T + U)(\vec{v}) + (T + U)(\vec{w}).\end{aligned}$$

So $L(\mathbb{R}^n)$ is closed under addition. Moreover, the linear transformation $\vec{v} \mapsto \vec{0}$ is the identity, and for any T the inverse is $-T$. Since addition in $\mathbb{R}$ is associative, so is addition in $L(\mathbb{R}^n)$.

Some of the previous examples have the property that the group operation is commutative, that is $ab = ba$ for all $a, b \in G$. Groups with this property are called **abelian**. Named after Niel Abel, a noted Norwegian mathematician who studied such groups in the 1820's. Groups where there exist elements that do not commute are called **non-abelian**.

10.3 Further Examples of Groups

Now we'll present a few more examples of groups. These are the examples that will be important to us in this course, since we will use them quite frequently.

10.3.1 Symmetric and Alternating Groups

A *permutation* of a set X is a bijection $X \to X$. The set of *all* permutations of a set X, is a group under composition. This set is denoted by S_X and called it the **symmetric group of X**.

$$S_X = \{\alpha : X \to X \quad \mid \alpha \text{ is a bijection}\}.$$

In the case where X is the set $[n] = \{1, 2, 3, \ldots, n\}$ then we denoted $S_{[n]}$ simply by S_n, and called it the *symmetric group of degree n*.

The set of even permutations A_n in S_n is also a group. Since it is a subset of S_n we call it a *subgroup* of S_n.

For example, consider A_4: the set of even permutations of degree 4. We know $|A_4| = \frac{4!}{2} = 12$ and we can list all the permutations in A_4 as follows:

$\varepsilon = (1)$, $\sigma_1 = (1,2)(3,4)$, $\sigma_2 = (1,3)(2,4)$, $\sigma_3 = (1,4)(2,3)$, $\sigma_4 = (1,2,3)$, $\sigma_5 = (1,3,2)$, $\sigma_6 = (1,2,4)$, $\sigma_7 = (1,4,2)$, $\sigma_8 = (1,3,4)$, $\sigma_9 = (1,4,3)$, $\sigma_{10} = (2,3,4)$, $\sigma_{11} = (2,4,3)$.

We can compute all possible products of two elements of the group and display them in a multiplication table. This table contains all the information of the group A_4. For example, the inverse of σ_6 is σ_7 since ε appears as table entry $\sigma_6\sigma_7$. Also, A_4 is not abelian, since the table is not symmetric about the diagonal line ↘.

	ε	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7	σ_8	σ_9	σ_{10}	σ_{11}
ε	ε	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7	σ_8	σ_9	σ_{10}	σ_{11}
$(1,2)(3,4) = \sigma_1$	σ_1	ε	σ_3	σ_2	σ_8	σ_{10}	σ_9	σ_{11}	σ_4	σ_6	σ_5	σ_7
$(1,3)(2,4) = \sigma_2$	σ_2	σ_3	ε	σ_1	σ_{11}	σ_6	σ_5	σ_8	σ_7	σ_{10}	σ_9	σ_4
$(1,4)(2,3) = \sigma_3$	σ_3	σ_2	σ_1	ε	σ_7	σ_9	σ_{10}	σ_4	σ_{11}	σ_5	σ_6	σ_8
$(1,2,3) = \sigma_4$	σ_4	σ_{11}	σ_7	σ_8	σ_5	ε	σ_3	σ_{10}	σ_6	σ_1	σ_2	σ_9
$(1,3,2) = \sigma_5$	σ_5	σ_9	σ_{10}	σ_6	ε	σ_4	σ_8	σ_2	σ_3	σ_{11}	σ_7	σ_1
$(1,2,4) = \sigma_6$	σ_6	σ_{10}	σ_9	σ_5	σ_2	σ_{11}	σ_7	ε	σ_1	σ_4	σ_8	σ_3
$(1,4,2) = \sigma_7$	σ_7	σ_8	σ_4	σ_{11}	σ_9	σ_3	ε	σ_6	σ_{10}	σ_2	σ_1	σ_5
$(1,3,4) = \sigma_8$	σ_8	σ_7	σ_{11}	σ_4	σ_{10}	σ_1	σ_2	σ_5	σ_9	ε	σ_3	σ_6
$(1,4,3) = \sigma_9$	σ_9	σ_5	σ_6	σ_{10}	σ_3	σ_7	σ_{11}	σ_1	ε	σ_8	σ_4	σ_2
$(2,3,4) = \sigma_{10}$	σ_{10}	σ_6	σ_5	σ_9	σ_1	σ_8	σ_4	σ_3	σ_2	σ_7	σ_{11}	ε
$(2,4,3) = \sigma_{11}$	σ_{11}	σ_4	σ_8	σ_7	σ_6	σ_2	σ_1	σ_9	σ_5	σ_3	ε	σ_{10}

We can use Sage to construct multiplication tables. The command to use is `cayley_table()`.

Sage

```
sage: A4=AlternatingGroup(4)
sage: A4.cayley_table()
 *  a b c d e f g h i j k l
 +------------------------
a| a b c d e f g h i j k l
b| b c a f d e h i g l j k
c| c a b e f d i g h k l j
d| d g j a h k b e l c f i
e| e i k c g l a f j b d h
f| f h l b i j c d k a e g
```

```
g| g j d k a h e l b i c f
h| h l f j b i d k c g a e
i| i k e l c g f j a h b d
j| j d g h k a l b e f i c
k| k e i g l c j a f d h b
l| l f h i j b k c d e g a
```

Notice that we have no idea which element of A_4 each letter represents. We can use the command `column_keys()` to find out.

Sage

```
sage: A4.cayley_table().column_keys()
((), (2,3,4), (2,4,3), (1,2)(3,4), (1,2,3), (1,2,4), (1,3,2), (1,3,4),
(1,3)(2,4), (1,4,2), (1,4,3), (1,4)(2,3))
```

This tells us the order the elements appear in the column and row headings. In other words, $a = ()$, $b = (2,3,4)$, etc. We can change the order of the elements in the table by creating a list with the order we want, then passing the list to `cayley_table()` using the optional argument `elements=`.

Sage

```
sage: A4list=["()", "(1,2)(3,4)", "(1,3)(2,4)", "(1,4)(2,3)", "(1,2,3)", "(1,3,2)",
"(1,2,4)", "(1,4,2)", "(1,3,4)", "(1,4,3)", "(2,3,4)", "(2,4,3)"]
sage: A4.cayley_table(elements=A4list)
*  a b c d e f g h i j k l
 +------------------------
a| a b c d e f g h i j k l
b| b a d c i k j l e g f h
c| c d a b l g f i h k j e
d| d c b a h j k e l f g i
e| e l h i f a d k g b c j
f| f j k g a e i c d l h b
g| g k j f c l h a b e i d
h| h i e l j d a g k c b f
i| i h l e k b c f j a d g
j| j f g k d h l b a i e c
k| k g f j b i e d c h l a
l| l e i h g c b j f d a k
```

We can also change the names it uses to represent the elements. We first create a list of "names", in precisely the same order as our elements are listed in `A4list`, then pass this to `cayley_table()` using the optional argument `names=`.

Sage

```
sage: A4names=["1", "s1", "s2", "s3", "s4", "s5", "s6", "s7", "s8", "s9",
 "s10", "s11"]
sage: A4.cayley_table(names=A4names,elements=A4list)
 *    1  s1  s2  s3  s4  s5  s6  s7  s8  s9 s10 s11
   +------------------------------------------------
  1|   1  s1  s2  s3  s4  s5  s6  s7  s8  s9 s10 s11
 s1|  s1   1  s3  s2  s8 s10  s9 s11  s4  s6  s5  s7
 s2|  s2  s3   1  s1 s11  s6  s5  s8  s7 s10  s9  s4
 s3|  s3  s2  s1   1  s7  s9 s10  s4 s11  s5  s6  s8
 s4|  s4 s11  s7  s8  s5   1  s3 s10  s6  s1  s2  s9
 s5|  s5  s9 s10  s6   1  s4  s8  s2  s3 s11  s7  s1
 s6|  s6 s10  s9  s5  s2 s11  s7   1  s1  s4  s8  s3
 s7|  s7  s8  s4 s11  s9  s3   1  s6 s10  s2  s1  s5
 s8|  s8  s7 s11  s4 s10  s1  s2  s5  s9   1  s3  s6
 s9|  s9   s5   s6 s10   s3   s7 s11   s1    1   s8   s4   s2
s10| s10   s6   s5   s9   s1   s8   s4   s3   s2   s7 s11    1
s11| s11   s4   s8   s7   s6   s2   s1   s9   s5   s3    1 s10
```

And there is our multiplication table, labeled exactly how we wanted.

Exercise 10.1: Construct a multiplication table for S_3. First list the elements of S_3 then work out the table. Check your resulting table by using Sage.

Answer on page 123

10.3.2 Finite Cyclic Groups

Consider the set of Rubik's cube moves $G = \{\varepsilon, \mathrm{R}, \mathrm{R}^2, \mathrm{R}^3\}$. Notice that the composition of any moves in this set is still in this set. For example, move R followed by move R^2 is move R^3, similarly move R^3 followed by move R^2 is move R. In other words,

$$\mathrm{R}\mathrm{R}^2 = \mathrm{R}^3, \quad \text{and} \quad \mathrm{R}^3\mathrm{R}^2 = \mathrm{R}.$$

Each element has an inverse, $\mathrm{R}^{-1} = \mathrm{R}^3$ and $(\mathrm{R}^2)^{-1} = \mathrm{R}^2$.

It follows that this set G is a group. It has the particular property that every element in G is some power of R (even the identity is a power of R: $\varepsilon = \mathrm{R}^0 = \mathrm{R}^4$). A group with this property is called a *cyclic group*.

Definition 10.3 (Cyclic Group): A group G is called **cyclic** if there is one element in G, say g, so that every other element of G is a power of g:

$$G = \{g^k \mid k \in \mathbb{Z}\}.$$

In this case we write $G = \langle g \rangle$, and say g is a **generator** for G.
If g has order n then $G = \{e, g, g^2, g^3, \ldots, g^{n-1}\}$ and we say G is a **cyclic group of order** n.

In the case when the group operation is addition then G is cyclic if every other element in a *multiple* of g: $G = \{kg \mid k \in \mathbb{Z}\}$.

In our example, G is a cyclic group of order 4, since it has four elements, and it is generated by R.

The multiplication table for G is

G	ε	R	R^2	R^3
ε	ε	R	R^2	R^3
R	R	R^2	R^3	ε
R^2	R^2	R^3	ε	R
R^3	R^3	ε	R	R^2

As another example consider the move sequence $\alpha = R^2U^2$ of the Rubik's cube. This move has order 6, and if we consider the set of all powers of this move, we get a cyclic group of order 6:

$$H = \langle \alpha \rangle = \{\varepsilon, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5\}.$$

The multiplication table for H is

H	ε	α	α^2	α^3	α^4	α^5
ε	ε	α	α^2	α^3	α^4	α^5
α	α	α^2	α^3	α^4	α^5	ε
α^2	α^2	α^3	α^4	α^5	ε	α
α^3	α^3	α^4	α^5	ε	α	α^2
α^4	α^4	α^5	ε	α	α^2	α^3
α^5	α^5	ε	α	α^2	α^3	α^4

You may have noticed that in each of our examples all elements commute under the operation. In other words, the group is abelian. This is true for any cyclic group.

Definition 10.4 (Cyclic Groups are Abelian): Let G be a cycle group. For any $a, b \in G$, $ab = ba$.

Proof: Let $G = \langle g \rangle$. For $a, b \in G$ there exist r and s such that $a = g^r$ and $b = g^s$, and so $ab = g^r g^s = g^{r+s} = g^{s+r} = g^s g^r = ba$.

□

In the examples above each element is determined precisely by the power of R (or α), so let's write out the multiplication table where we just write i, in place of R^i (or α^i).

G	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

H	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	2	3	4	5	0
2	2	3	4	5	0	1
3	3	4	5	0	1	2
4	4	5	0	1	2	3
5	5	0	1	2	3	4

These tables represent the way we "multiply" in G and H. If we look closely we see that to multiply α^2 and α^4 we just add the exponents, and if the sum is larger than 5 then we take the remainder when divided by 6. So in this case $2 + 4 = 6$ which has remainder 0 when divided by 6.

In the next section, we investigate this "remainder" operation on the set of integers.

10.3.3 Group of Integers Modulo n: $\mathbb{Z}_n$

Consider the set $\mathbb{Z}_{12} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$. We define the operation $+_{12}$ to be *addition modulo* 12. By this we mean $a +_{12} b$ is the remainder of $a + b$ when divided by 12. This type of addition is familiar to anyone who adds time on a clock. For example, if it is 8-o'clock, then 6 hours later is $8 +_{12} 6 = 2$, or 2-o'clock.

Some examples are: $2 +_{12} 3 = 5$, $7 +_{12} 5 = 0$, since $7 + 5 = 12$ which is divisible by 12, and $11 +_{12} 10 = 9$, since $11 + 10 = 21$ which has remainder 9 when divided by 12. In Sage remainders are computed using the operator %.

Sage

```
sage: (2+3)%12
5
sage: (7+5)%12
0
sage: (11+10)%12
9
```

Is $\mathbb{Z}_{12}$ a group under this "new" addition $+_{12}$? Let's check the properties one-by-one.

closed: Since the remainder will always be a number between 0 and 11 then $\mathbb{Z}_{12}$ is certainly closed under $+_{12}$.

associative: This addition is associative, since it is built from regular addition of integers which is associative.

identity: The identity is 0, since $0 +_{12} b = b$ for all $b \in \mathbb{Z}_{12}$.

inverses: What is the inverse of an element? For example, what is the inverse of 3? This would be a number b such that 12 divides $3 + b$. The number $12 - 3 = 9$ has this property. So the inverse of 3 is 9. In general, the inverse of a is $12 - a$.

It follows that $\mathbb{Z}_{12}$ is a group.

There was nothing special about 12 in this example, other than it being familiar to us from our experience dealing with clocks. We can really do this for any positive integer n.

Definition 10.5: Let $n > 1$ be and integer. Define an operation on the set $\mathbb{Z}_n = \{0, 1, 2, 3, \ldots, n-1\}$, called *addition modulo n*, as follows. For $a, b \in \mathbb{Z}_n$, let $a +_n b$ be the remainder of $a + b$ when divided by n. $\mathbb{Z}_n$ is a group under addition modulo n, and is called the (additive) **group of integers modulo** $\boldsymbol{n}$. Since this group is cyclic it is often called the (additive) **cyclic group of order** $\boldsymbol{n}$.

The group $\mathbb{Z}_n$ is also usually denoted by $\mathbb{Z}/n\mathbb{Z}$, which is read "$\mathbb{Z}$ mod n $\mathbb{Z}$".

Why is $\mathbb{Z}_n$ cyclic? Each element of $\mathbb{Z}_n$ can be obtained from 1 by repeatedly adding 1 to itself. Note, our group operation is addition so the analogy of a "power" is a multiple. Since every element of $\mathbb{Z}_n$ is a suitable multiple of 1 then $\mathbb{Z}_n = \langle 1 \rangle$.

Notation & Terminology:

If a, b, and n are integers we say a **is congruent to** b **modulo** n if $n \mid b - a$ and we write $a \equiv b \mod n$. For example, $15 \equiv 3 \mod 12$, and $6 \equiv 2 \mod 4$, but $7 \not\equiv 3 \mod 5$ since $5 \not| \, 7 - 3$.

Addition of two integers, a and b, modulo n, which we denoted as $a +_n b$ is often denoted by

$$a + b \pmod{n}.$$

For example, $11 +_{12} 10 = 9$ could also be written as $11 + 10 \equiv 9 \pmod{12}$.

In Section 10.3.2 we saw the multiplication tables for G and H, written only using the exponents, are precisely the groups $\mathbb{Z}_4$ and $\mathbb{Z}_6$. This observation, is true in general, in the sense that *every finite cyclic group is essentially* $\mathbb{Z}_n$ *for some integer* n. The only difference it just how the elements were named, which is superficial.

Finite cyclic groups are built into Sage with the command `CyclicPermutationGroup()`. As the name suggests, cyclic groups are constructed using permutations. Lets look at an example.

Sage

```
sage: Z5=CyclicPermutationGroup(5)
sage: Z5.list()
[(), (1,2,3,4,5), (1,3,5,2,4), (1,4,2,5,3), (1,5,4,3,2)]
```

Here, $\mathbb{Z}_5$ is represented by using the 5-cycle $(1, 2, 3, 4, 5)$ as a generator. We can compute the multiplication table by first telling Sage how to name the elements.

```
Sage
sage: Z5list=["()", "(1,2,3,4,5)", "(1,3,5,2,4)", "(1,4,2,5,3)", "(1,5,4,3,2)"]
sage: Z5names=["0","1","2","3","4"]
sage: Z5.cayley_table(names=Z5names,elements=Z5list])
*  0 1 2 3 4
 +----------
0| 0 1 2 3 4
1| 1 2 3 4 0
2| 2 3 4 0 1
3| 3 4 0 1 2
4| 4 0 1 2 3
```

If one wants to work with $\mathbb{Z}_n$ where the elements are $\{0, 1, \ldots, n-1\}$, rather than permutations, then this can be done using `IntegerModRing()`. Though, for just doing calculations we would use the modulo operator `%`, as in the clock example above.

```
Sage
sage: Z5=IntegerModRing(5)
sage: Z5.list()
[0, 1, 2, 3, 4]
sage: Z5(3)+Z5(4)
2
```

Exercise 10.2: Construct a Cayley table for $\mathbb{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}$, under addition modulo 7. Check your results using Sage.

Answer on page 123

Example 10.1: We determine the order of each element in $\mathbb{Z}_{12} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$. 1 has order 12. Since $6 \cdot 2 \equiv 2+2+2+2+2+2 \equiv 0 \pmod{12}$ then 2 has order 6. Similarly $4 \cdot 3 \equiv 0 \pmod{12}$ so 3 has order 4. Continuing in this way we find:

k	**elements of order k**
1	0
2	6
3	$4, 8$
4	$3, 9$
6	$2, 10$
12	$1, 5, 7, 11$

It follows that $1, 5, 7$, and 11 are all generators of $\mathbb{Z}_{12}$. That is,

$$\mathbb{Z}_{12} = \langle 1 \rangle = \langle 5 \rangle = \langle 7 \rangle = \langle 11 \rangle.$$

A few curious things to note: (i) the only orders that show up are divisors of 12, and (ii) the generators of $\mathbb{Z}_{12}$ are the elements relatively prime to 12. Are these coincidences? We'll see in Corollary 11.4 and Theorem 11.9 that these observations are not coincidences.

10.3.4 Group of Units Modulo n: $U(n)$

You may wonder if we can do the same thing with multiplication, instead of addition, on $\mathbb{Z}_n$. That is, does $\mathbb{Z}_n$ form a group under *multiplication modulo n*, which we denote by $\cdot_n$?

First we notice that the identity would be 1, but of course 0 doesn't have a (multiplicative) inverse. So let's take 0 out of consideration, and just focus on the set $\mathbb{Z}_n^* = \{1, 2, 3, \ldots, n-1\}$.

As an example consider $\mathbb{Z}_6^* = \{1, 2, 3, 4, 5\}$. Lets check to see if this set is closed under multiplication modulo 6. Well, $3 \cdot_6 5 = 3 \in \mathbb{Z}_6^*$, so far so good. But $2 \cdot_6 3 = 0 \notin \mathbb{Z}_6^*$. Therefore, $\mathbb{Z}_6^*$ is definitely not closed under multiplication, so it is *not a group*.

But all is not lost. It just seems that some elements in $\mathbb{Z}_6^*$ are just trouble-makers. Their presence prevents it from being closed under multiplication. Who are these trouble makers? Lets find out.

$$\begin{array}{llll}
1 \cdot_6 2 = 2 & 1 \cdot_6 3 = 3 & 1 \cdot_6 4 = 4 & 1 \cdot_6 5 = 5 \\
2 \cdot_6 2 = 4 & 2 \cdot_6 3 = 0 \notin \mathbb{Z}_6^* & 2 \cdot_6 4 = 2 & 2 \cdot_6 5 = 4 \\
3 \cdot_6 3 = 3 & 3 \cdot_6 4 = 0 \notin \mathbb{Z}_6^* & 3 \cdot_6 5 = 3 & 4 \cdot_6 4 = 4 \\
4 \cdot_6 5 = 2 & 5 \cdot_6 5 = 1 & &
\end{array}$$

The elements 2, 3 and 4 seem to be causing the problems. These are precisely the elements that have a factor in common with 6. Is this a coincidence? Not at all, the remainder of division by 6 will always be between 0 and 5, and since $\mathbb{Z}_6^*$ does not contain 0, the trouble makers are the numbers whose products are divisible by 6. For two numbers $a, b \in \mathbb{Z}_6^*$ to have a product divisible by 6, they each must have a factor in common with 6.

We say two numbers are **relatively prime** if they do not have a common prime factor. If two numbers have a common factor then we say they are *not relatively prime*. Note that if two numbers are relatively prime, then they have no common prime factor, and so their greatest common divisor is 1. This means a and b are relatively prime if and only if $\gcd(a, b) = 1$. See Appendix B for more information on greatest common divisors.

We have just determined that the trouble makers are the numbers which are not relatively prime to 6. Namely, 2, 3, and 4.

Therefore, consider just the set of numbers in $\mathbb{Z}_6^*$ that are relatively prime to 6: This set is denoted by $U(6)$:

$$U(6) = \{1, 5\}.$$

This set is a group! The inverse of 5 is itself. The multiplication table is:

U(6)	1	5
1	1	5
5	5	1

Sage

```
sage: U6=[m for m in range(0,6) if gcd(m,6)==1]
{1, 5}
```

The previous construction can be done for any integer n in place of 6. This is the next definition.

Definition 10.6 (Group of Units Modulo n): Let $n > 1$ be and integer, and let

$$U(n) = \{m \mid 1 \le m \le n-1 \text{ and } \gcd(m, n) = 1\}.$$

$U(n)$ is a group under multiplication modulo n, and is called the **group of units modulo** n. In the case when p is prime, $U(p) = \mathbb{Z}_p^* = \{1, 2, 3, \ldots, p-1\}$.

The number of elements in $U(n)$ is precisely the integers between 1 and n which are relatively prime to n. There is an important number-theoretic function, called *Euler's ϕ-function*, denoted by ϕ, which calculates this number: $\phi(n) = |U(n)|$. See Section B.3 in Appendix B for information on this function including a simple formula to calculate its value from the prime factorization of n.

Euler's ϕ-function is implemented in Sage, under the command `euler_phi()`. For example, here we see $\phi(6) = 2$.

Sage
```
sage: euler_phi(6)
2
```

Exercise 10.3: Determine the elements of the set $U(8)$, and construct the multiplication table.

Answer on page 123

Example 10.2: In this example we will investigate the group $U(18)$, which has 6 elements.

Sage
```
sage: euler_phi(18)
6
```

Of course, we could have done this by hand (or use Sage code similar to the example we did for $U(6)$). We would just go through the numbers from 1 to 18 and omit any that have a factor of 2 or 3.

$$U(18) = \{1, 5, 7, 11, 13, 17\}.$$

What is the inverse of 11? One way is to compute the product of 11 with each element of $U(18)$ and check when we get 1:

Sage
```
sage: for m in [1, 5, 7, 11, 13, 17]:
sage:     if 11*m%18==1:
sage:        print m
5
```

Therefore $11^{-1} = 5$ in $U(18)$.

A more efficient way to find the inverse is to use the Extended Euclidean Algorithm: If a and b are integers and $gcd(a, b) = d$ then there must be integers u and v so that $ua + vb = d$ (see Theorem B.4 in Appendix B). The standard algorithm for finding the $\gcd$ is called the *Euclidean Algorithm*, and the algorithm for producing numbers u and v is called the *Extended Euclidean Algorithm*. For details of these algorithms see Appendix B. These algorithms are implemented in Sage, so we can use them.

Sage
```
sage:  d,u,v = xgcd(11,18)
sage:  print u,v
5, -3
```

How does this help us find 11^{-1}? Well, the Extended Euclidean Algorithm has returned three numbers: the first is 1 which is the $\gcd$, the other two, 5 and -3, have the property that $5(11) + (-3)(18) = 1$. This means $5(11)$ has remainder 1 when divided by 18. Which is exactly what it means for 5 to be an inverse of 11.

To find the inverse of 13 we can do the same thing, and get $13^{-1} = 7$.

Sage
```
sage:  d,u,v = xgcd(13,18)
sage:  print u,v
7, -5
```

We can write a function called `inverse` that will return the inverse of a in $U(m)$.

Sage

```
sage: def inverse(a,m):
sage:     d,u,v=xgcd(a,m)
sage:     if d==1:
sage:         return u%m           # return inverse as a number between 1 and m-1
sage:     else:
sage:         return a, "is not in U group"     # just in case a is not in U(m)

sage:  inverse(11,18)
5
sage:  inverse(13,18)
13
```

Actually, Sage already has a built in inverse function of $U(n)$. The syntax is `inverse_mod(a,n)` and it returns the inverse of a in $U(n)$.

To compute the order of an element, we can take successive powers until we hit the identity. As an example, we determine the order of 11 is 6.

Sage

```
sage:  for n in (1..6):
sage:      print n, 11^n%18
1 11
2 13
3 17
4 7
5 5
6 1
```

We can also create a function to do this. We'll see next lecture that the order of an element must divide the order of the group so we can limit the exponents we need to check. The function `divisors(m)` returns a list of the divisors of m, arranged from smallest to largest. Recall $|U(m)| = \phi(m)$, the Euler ϕ-function.

Sage

```
sage:   def order(a,m):
sage:      if gcd(a,m)==1:              # first check that a is in U(m)
sage:          for k in divisors(euler_phi(m)):    # order must divide |U(m)|
sage:              if a^k%m==1:
sage:                  return k
sage:      else:
sage:          return a, "is not in U group"

sage:  order(5,18)
6
sage:  order(13,18)
3
```

It follows that $U(18)$ is a cyclic group generated by 5:

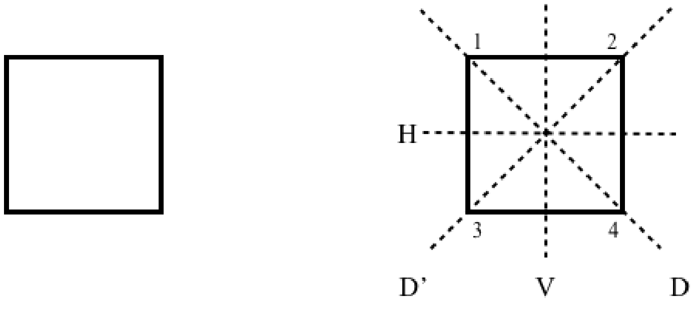
$$U(18) = \langle 5 \rangle.$$

The element 11 also generates the group.

$U(18)$ has proper subgroups $\{1\}$, $\{1, 17\}$, and $\{1, 7, 13\}$.

10.3.5 Dihedral Groups: D_n

Consider a square as shown in Figure 10.1a. We want to determine all the ways we can pick up the square, move it in some way, then put it back in the original space it occupied. If an observer didn't see us pick it up, but only saw it before and after, they shouldn't notice any change. For example, we could rotate it 90 degrees, or we could flip it over a horizontal line. We'd like to determine all possible ways we could have moved the square. In some sense, the number of ways we can do this is related to how "symmetric" a square is.



(a) (b)

Figure 10.1: A square and its lines of symmetry.

Let G denote the set of ways in which we can move the square. To keep track of the motions, we can label the vertices of the squares as $1, 2, 3, 4$, see Figure 10.1b, and each motion corresponds to a permutation of the labels on the vertices. In Table 10.1 we list the elements of G: $G = \{R_0, R_{90}, R_{180}, R_{270}, H, V, D, D'\}$.

notation	**description**	**permutation**
R_0	rotation of $0°$ (i.e. do nothing)	ε
R_{90}	rotation of $90°$ (clockwise)	$(1,2,3,4)$
R_{180}	rotation of $180°$ (clockwise)	$(1,3)(2,4)$
R_{270}	rotation of $270°$ (clockwise)	$(1,4,3,2)$
H	reflection of $180°$ about horizontal axis	$(1,4)(2,3)$
V	reflection of $180°$ about vertical axis	$(1,2)(3,4)$
D	reflection of $180°$ about diagonal axis (see diagram below)	$(2,4)$
D'	reflection of $180°$ about other diagonal axis (see diagram below)	$(1,3)$

Table 10.1: Symmetries of the square

We can combine elements of G through consecutive motions. For example, $R_{90}H$ means first rotate by $90°$, then reflect about the horizontal axis. The resulting motion is equivalent to D'. We can see this by actually doing both motions $R_{90}H$ and D' and observing they do exactly the same thing. Or we could compose their corresponding permutations: $(1,2,3,4)(1,4)(2,3) = (1,3)$.

G is a group under this way of composing moves. It is the *group of symmetries of the square*, or more commonly called **the dihedral group of order 8**, and denoted by D_4. The multiplication table for D_4 is

D_4	R_0	R_{90}	R_{180}	R_{270}	H	V	D	D'
R_0	R_0	R_{90}	R_{180}	R_{270}	H	V	D	D'
R_{90}	R_{90}	R_{180}	R_{270}	R_0	D'	D	H	V
R_{180}	R_{180}	R_{270}	R_0	R_{90}	V	H	D'	D
R_{270}	R_{270}	R_0	R_{90}	R_{180}	D	D'	V	H
H	H	D	V	D'	R_0	R_{180}	R_{90}	R_{270}
V	V	D'	H	D	R_{180}	R_0	R_{270}	R_{90}
D	D	V	D'	H	R_{270}	R_{90}	R_0	R_{180}
D'	D'	H	D	V	R_{90}	R_{270}	R_{180}	R_0

The analysis carried out for a square can similarly be done for any regular n-gon, $\mathcal{P}_n$ (where $n \geq 3$). See Figure 10.2 for some familiar n-gons. If $n = 3$ then $\mathcal{P}_3$ is an equilateral triangle. If $n = 4$ then $\mathcal{P}_4$ is a square as we just considered. If $n = 5$ then $\mathcal{P}_5$ is a regular pentagon, and so on. The corresponding group is denoted by D_n and is called the **dihedral group of order** $2n$.

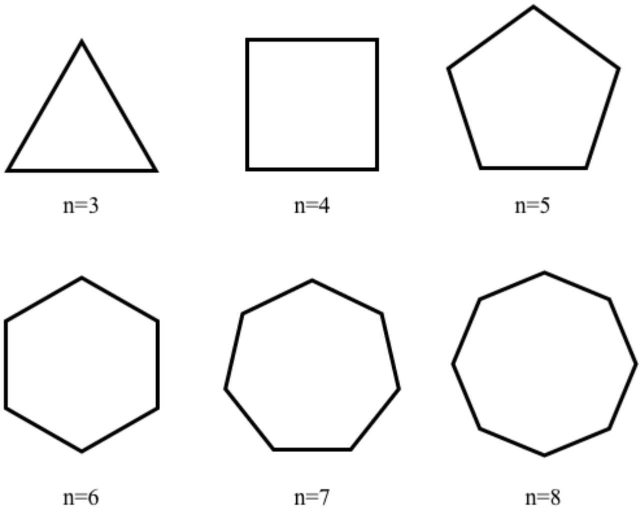


Figure 10.2: Some regular n-gons.

Dihedral groups are frequently found in art and nature, and they are a very important type of group used by mineralogists to study crystals.

You may wonder where the "2n" comes from in the name. Looking back at the square we see that there are 8 motions preserving the square (we call these the symmetries of the square). Four were rotations, and four were reflections. This is true for any regular n-gon. There will be n rotational symmetries and n reflective symmetries, for a total of $2n$.

Dihedral groups are built into Sage. Each element is represented as permutations of the vertices of the n-gon. Here is an example with D_4.

Sage

```
sage: D4=DihedralGroup(4)
sage: D4.list()          #lists the elements of D4 as represented in SAGE
[(), (2,4), (1,2)(3,4), (1,2,3,4), (1,3), (1,3)(2,4), (1,4,3,2),
(1,4)(2,3)]
```

We can assign each element to a name. For example, $(1, 2, 3, 4)$ corresponds to the $90°$ rotation R_{90}.

Sage

```
sage: R90=D4("(1,2,3,4)")
sage: R180=D4("(1,3)(2,4)")
```

```
sage: R270=D4("(1,4,3,2)")
sage: H=D4("(1,4)(2,3)")
sage: V=D4("(1,2)(3,4)")
sage: D=D4("(2,4)")
sage: Dp=D4("(1,3)")        # we use Dp for D'
```

We can now compute products. For example, we see $R_{90}D = H$.

Sage

```
sage: R90*D
(1,4)(2,3)
```

The full multiplication table for D_4 can be computed in Sage as follows.

Sage

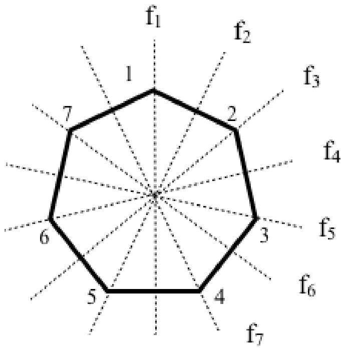
```
sage: D4list=["()", "(1,2,3,4)", "(1,3)(2,4)", "(1,4,3,2)", "(1,4)(2,3)",
"(1,2)(3,4)", "(2,4)", "(1,3)"]
sage: D4names=["R0","R90","R180","R270","H","V","D","D'"]
sage: D4.cayley_table(names=D4names,elements=D4list)

*    R0  R90 R180 R270    H    V    D   D'
    +----------------------------------------
  R0|   R0  R90 R180 R270    H    V    D   D'
 R90|  R90 R180 R270   R0   D'    D    H    V
R180| R180 R270   R0  R90    V    H   D'    D
R270| R270   R0  R90 R180    D   D'    V    H
   H|    H    D    V   D'   R0 R180  R90 R270
   V|    V   D'    H    D R180   R0 R270  R90
   D|    D    V   D'    H R270  R90   R0 R180
  D'|   D'    H    D    V  R90 R270 R180   R0
```

10.3.6 Notation for D_n

For a regular n-gon we typically use r to denote a clockwise rotation through $\frac{360}{n}$ degrees, and more generally, r^k to denote a clockwise rotation through $k\frac{360}{n}$ degrees. A reflection through a line of symmetry is denoted by f_i, for $1 \leq i \leq n$.

For example, the lines of symmetry for a regular 7-gon are labelled below. Some of the elements are described in Table 10.2. There are 14 elements in D_7:

$$D_7 = \{1, r, r^2, r^3, r^4, r^5, r^6, f_1, f_2, f_3, f_4, f_5, f_6, f_7\}.$$



notation	**description**	**permutation**
1	rotation of $0°$ (i.e. do nothing)	ε
r	rotation of $\frac{360}{7}$ degrees (clockwise)	$(1,2,3,4,5,6,7)$
r^k	rotation of $k\frac{360}{n}$ degrees (clockwise) for $1 \leq k \leq 7$	7-cycle
f_1	reflection of $180°$ about f_1 line	$(2,7)(3,6)(4,5)$
f_i	reflection of $180°$ about f_i axis for $1 \leq i \leq 7$	product of three 2-cycles

Table 10.2: Symmetries of a regular 7-gon

One can check that every element of D_7 can be expressed as a product of the form $r^k f_1^\ell$ for some $0 \leq k \leq 6$, and $0 \leq \ell \leq 1$. For example, $f_5 = r^3 f_1$. We say D_7 is generated by r, f_1 and write

$$D_7 = \langle r, f_1 \rangle.$$

10.4 Exercises

1. Give two reasons why the set of odd integers under addition is not a group.

2. Show that $\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$ does not have a multiplicative inverse in $GL(2,\mathbb{R})$.

3. Show that the group $GL(2,\mathbb{R})$ is non-abelian by finding two matrices A and B in $GL(2,\mathbb{R})$ where $AB \neq BA$.

4. Find the inverse of $\begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix}$ in $SL(2,\mathbb{R})$. First verify it is in $SL(2,\mathbb{R})$.

5. The group operation $*$ is frequently omitted, for example $a * b$ would just be written as ab. This is due to the fact that we often refer to the operation as "multiplication". However, when the operation is addition we keep the $+$ symbol, and we also use 0 for the identity instead of e. Translate each of the following multiplicative expression into its additive counterpart.

   (a) $a^2 b$

   (b) $b^4 a^{-3} b$

   (c) $(ab^3)^{-2} c^3 = e$

6. Let $G = \{a, b, c, d\}$ have an operation $*$ with corresponding multiplication table

$*$	a	b	c	d
a	a	b	c	d
b	b	a	d	c
c	c	d	a	b
d	d	d	b	c

   Is G a group under this operation? Explain.

Dihedral Groups:

Exercises 7 through 13 investigate the dihedral groups.

7. (a) With pictures and words, describe each symmetry in D_3 (the set of symmetries of an equilateral triangle).
   (b) Write out a complete multiplication (Cayley) table for D_3.
   (c) Is D_3 abelian (that is, does every element commute with every other element)?

8. With pictures and words, describe each symmetry in D_5 (the set of symmetries of a regular pentagon).

9. For $n \geq 3$ describe the elements of D_n. (You will need to consider two cases, depending on whether n is even or odd.)

10. In D_n, explain geometrically why

    (a) a reflection followed by a reflection must be a rotation.
    (b) a reflection and a rotation taken together in either order must be a reflection.

11. Is D_n a cyclic group? That is, does $D_n = \langle g \rangle$ for some $g \in D_n$?

12. Is D_n abelian?

13. If r_1, r_2, and r_3 represent rotations and f_1, f_2, and f_3 represent reflections from D_n, determine whether $f_1 r_3 r_2 f_2 r_1 f_1$ is a rotation or reflection.

Group of Integers under addition modulo n:

Exercises 14 through 18 investigate the group of integers modulo n: $\mathbb{Z}_n$.

14. List the element of $\mathbb{Z}_2$, and write out a multiplication table for this group.

15. Determine the following in $\mathbb{Z}_{15}$

    (a) $7 +_{15} 6$
    (b) $13 +_{15} 8$
    (c) $12 \cdot 7$
    (d) the inverse of 11
    (e) the inverse of 3
    (f) ord(10)
    (g) ord(7)

16. Determine the order of each element in $\mathbb{Z}_{10}$.

17. Determine which elements of $\mathbb{Z}_{10}$ are generators for $\mathbb{Z}_{10}$. That is, find all $g \in \mathbb{Z}_{10}$ such that $\mathbb{Z}_{10} = \langle g \rangle$.

18. Find all the elements of $g \in \mathbb{Z}_{18}$ for which $\mathbb{Z}_{18} = \langle g \rangle$.

Unit Group modulo n:

Exercises 19 through 22 investigate the Unit Groups $U(n)$.

19. Determine the elements of the set $U(5)$, and construct the multiplication table.

20. Determine the elements of the set $U(12)$, and construct the multiplication table.

21. (a) How many elements does $U(37)$ have?
    (b) Find the inverse of 25 in $U(37)$.
    (c) What is the order of 25.
    (d) Is $U(37)$ cyclic? If so, find a generator.
    (Hint: use Sage to help with calculations.)

22. Is $U(20)$ cyclic?

Groups in General:

Exercises 23 through 31 investigate groups in general. Solutions to these exercises should be based on the four properties listed in the definition of a group, and any theorems which were consequences of these properties.

23. For any elements a and b from a group G, and any integer n, prove that $(b^{-1}ab)^n = b^{-1}a^nb$. (We've already shown this for permutations, so this question is asking you to verity this is really just a consequence of group properties.)

24. Let a and b be elements of an abelian group G, and let n be any integer. Show that $(ab)^n = a^nb^n$. Is this true for non-abelian groups? Explain.

25. If $a, b \in G$ such that ord$(a^2) =$ ord(b^2), is it necessarily true that ord$(a) =$ ord(b)?

26. In a group G show that the number of nonidentity elements that satisfy the equation $x^5 = e$ is a multiple of 4.

27. Show that if G is a group and $a \in G$ such that $a^2 = a$ then a must be the identity.

28. Suppose $G = \{e, a, b, c, d\}$ is a group with multiplication table

	e	a	b	c	d
e	e				
a		b			e
b		c	d	e	
c		d		a	b
d					

Fill in the blank entries.

29. Suppose $G = \{e, a, b, c, d, f\}$ is a group with multiplication table

	e	a	b	c	d	f
e	e	a	b	c	d	f
a	a	e				
b	b	f				
c	c			e	a	
d	d	c	a			
f	f	b	c	a	e	

Fill in the blank entries.

30. Prove Lemma 10.3.
(Hint: The first two parts are really just consequences of the left- and right- cancellation properties.)

31. Prove that if G is a group with the property that the square of every element is the identity (i.e. every element has order 2), then G is abelian.

32. Let G be a group with operation $\cdot$. For which operation $*$ is the set G a group under $*$?

    (a) $a * b = b \cdot a$

    (b) $a * b = b^{-1} \cdot a \cdot b$

    (c) $a * b = b^{-1} \cdot a$

    (d) $a * b = (a \cdot b)^2$

A few more examples of groups:

33. The integers 5 and 15 are among a collection of 12 integers that form a group under multiplication modulo 56. List all 12.

34. **Nim Group.** Consider the set $G = \{0, 1, 2, 3, 4, 5, 6, 7\}$. Suppose there is a group operation $*$ on G that satisfies the following two conditions:

(a) $a * b \leq a + b$ for all a, b in G,

(b) $a * a = 0$ for all a in G.

Construct the multiplication table for G. This groups is sometimes called the *Nim Group* due to its relationship to the game of Nim.

35. Prove that the set of all 3×3 matrices with real entries of the form

$$\begin{bmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{bmatrix}$$

is a group under matrix multiplication. (This group, sometimes called the *Heisenberg group* after the Nobel Prize winning physicist Werner Heisenberg, is intimately related to the Heisenberg Uncertainty Principle of Quantum Physics.)

Answers to in-chapter exercises:

Exercise 10.1:

	ε	$(1,2)$	$(1,3)$	$(2,3)$	$(1,2,3)$	$(1,3,2)$
ε	ε	$(1,2)$	$(1,3)$	$(2,3)$	$(1,2,3)$	$(1,3,2)$
$(1,2)$	$(1,2)$	ε	$(1,2,3)$	$(1,3,2)$	$(1,3)$	$(2,3)$
$(1,3)$	$(1,3)$	$(1,3,2)$	ε	$(1,2,3)$	$(2,3)$	$(1,2)$
$(2,3)$	$(2,3)$	$(1,2,3)$	$(1,3,2)$	ε	$(1,2)$	$(1,3)$
$(1,2,3)$	$(1,2,3)$	$(2,3)$	$(1,2)$	$(1,3,2)$	$(1,3)$	ε
$(1,3,2)$	$(1,3,2)$	$(1,3))$	$(2,3)$	$(1,2)$	ε	$(1,2,3)$

Exercise 10.2:

	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

Exercise 10.3: $U(8) = \{1, 3, 5, 7\}$

	1	3	5	7
1	1	3	5	7
3	3	1	7	5
5	5	7	1	3
7	7	5	3	1

Lecture 11

Subgroups

Last lecture we introduced the concept of a *group*. This is a set equipped with an (associative) operation that allows us to combine two elements to produce another element in the same set. We require the set, under this operation, to have an identity, and for every element to have an inverse. We saw a number of familiar sets and operations which satisfy the property of being a group. In this lecture we look at subsets of groups.

11.1 Subgroups

Not all subsets of groups are created equal. For example, consider the two subset of the set of all Rubik's cube moves: $H = \{\varepsilon, \mathrm{R}, \mathrm{R}^2, \mathrm{R}^3\}$ and $K = \{\varepsilon, \mathrm{R}, \mathrm{U}\}$. The set H is a group itself as we saw last lecture. On the other hand, the set K is not a group since, for one thing the product of R and U is not in K.

If G is a group, and H is a subset of G which is also a group (using the same operation), then we say H is a **subgroup** of G, and we write $H < G$.

Example 11.1: (a) $H = \{\varepsilon, (1,2)\}$ is a subgroup of S_3.

(b) The subset $G = \{\varepsilon, (1,2,3,4), (1,4,3,2), (1,3)(2,4), (1,4)(2,3), (1,2)(3,4), (2,4), (1,3)\}$ of S_4 is a subgroup. We could check that every product of elements in G is again in G, and that each element has an inverse in G. However, we could just observe that G is precisely D_4, the dihedral group of order 8 that we investigated last lecture, so we already know it is a group.

(c) The subset $\{0,2,4\}$ is a subgroup of the cyclic group of order 6, $\mathbb{Z}_6 = \{0,1,2,3,4,5\}$. It is closed since the sum of any two elements in $\{0,2,4\}$ is still in $\{0,2,4\}$. The inverse of 2 is 4 since $2+4 = 0 \pmod 6$.

To verify whether a subset of a group is itself a group we don't need to start from scratch. For instance, since the operation on G is associative, then restricting the operation to just elements of a subset H the operation would still have to be associative. This means we don't need to check associativity, we get this for free. So we really only need to check (i) H is closed, (ii) the identity is in H, and (iii) each element of H has an inverse in H. Notice that if we have (i) and (iii) then we get (ii) for free, since $aa^{-1} = e$. This means we have the following test for a subgroup.

Theorem 11.1 (Two-Step Subgroup Test): *Let G be a group and H a nonempty subset of G. If*

(a) for every $a, b \in H$, $ab \in H$ (closed under multiplication), and

(b) for every $a \in H$, $a^{-1} \in H$ (closed under inverses),

then H is a subgroup of G.

11.2 Examples of Subgroups

Imagine playing with Rubik's cube but only allowing yourself to use moves U and R. Some examples of move sequences that you could perform are: $\mathrm{RU\,R^{-1}\,U^2\,R^2\,U^{-1}}$, RURURURUR, and $(\mathrm{R^2U^2})^3$. Observe that every move sequence has an inverse involving only R and U, and the product of any two move sequences is another move sequence involving only R and U. That is, the set of all such move sequences is a group! We denote this group by $\langle \mathrm{R}, \mathrm{U} \rangle$.

For any group G, let $g_1, g_2, \ldots, g_k$ be elements in G. Let $\langle g_1, g_2, \ldots, g_k \rangle$ be the set of all elements of G which can be expressed as products of $g_1, g_2, \ldots, g_k$ and their inverses $g_1^{-1}, g_2^{-1}, \ldots, g_k^{-1}$:

$$\langle g_1, g_2, \ldots, g_k \rangle = \{x \in G \mid x = g_{j_i}^{m_i} g_{j_2}^{m_2} \cdots g_{j_\ell}^{m_\ell} \text{ for some indices } j_i\text{'s and exponents } m_i \in \mathbb{Z}\},$$

then $\langle g_1, g_2, \ldots, g_k \rangle$ is the **subgroup generated by** $g_1, g_2, \ldots, g_k$.

When $k = 1$, the group $\langle g \rangle = \{g^n \mid n \in \mathbb{Z}\}$ is called a **cyclic** subgroup of G.

Many of our examples of subgroups will be of these types, and this is how we will construct groups in Sage.

(1) Recall $S_3 = \{\varepsilon, (1,2), (1,3), (2,3), (1,2,3), (1,3,2)\}$.
One subgroup of S_3 is $\langle (1,2,3) \rangle = \{\varepsilon, (1,2,3), (1,3,2)\}$. Check this is indeed a subgroup.

We can list all subgroups of S_3 as follows:
$\langle \varepsilon \rangle = \{\varepsilon\}$
$\langle (1,2) \rangle = \{\varepsilon, (1,2)\}$
$\langle (1,3) \rangle = \{\varepsilon, (1,3)\}$
$\langle (2,3) \rangle = \{\varepsilon, (2,3)\}$
$\langle (1,2,3) \rangle = \{\varepsilon, (1,2,3), (1,3,2)\} = \langle (1,3,2) \rangle$

We can check that $\langle (1,2), (1,3) \rangle = S_3$. What this means is that any element of S_3 can be written as a product involving $(1,2)$ and $(1,3)$. In fact, the subgroup generated by *any two* elements will be all of S_3 again.

Sage

```
sage: S3=SymmetricGroup(3)
sage: a=S3("(1,2)")
sage: b=S3("(1,3)")
sage: H=PermutationGroup([a,b])    # forms the group generated by a and b
sage: H==S3   # check if H is equal to the whole group
true
```

(2) Recall $\mathbb{Z}_{10} = \{0,1,2,3,4,5,6,7,8,9,10\}$ and the operation is additional modulo 10. The subgroups of $\mathbb{Z}_{10}$ are:
$\langle 0 \rangle = \{0\}$
$\langle 2 \rangle = \{0,2,4,6,8\}$

$\langle 5\rangle = \{0, 5\}$.
There are no other (proper) subgroups of $\mathbb{Z}_{10}$.

(3) Recall $U(10) = \{1, 3, 7, 9\}$ and the operation is multiplication modulo 10. Since $3^2 = 9$ and $3^3 = 7$ then $U(10) = \langle 3\rangle$. A proper subgroup of $U(10)$ is $\langle 9\rangle = \{1, 9\}$. Verify this is the only other proper subgroup of $U(10)$, besides the trivial subgroup $\{1\}$.

(4) In S_{10}, the permutations $\alpha = (1, 2)$ and $\beta = (1, 5, 3)(2, 4)$ generate a subgroup H of size 120. The permutation $(1, 4, 3, 2)$ is in H since $\alpha\beta\alpha\beta^2 = (1, 4, 3, 2)$. On the other hand, $(8, 9, 10) \notin H$, since any product of α and β would have to fix 10.

Sage

```
sage: S10=SymmetricGroup(10)
sage: a=S10("(1,2)")
sage: b=S10("(1,5,3)(2,4)")
sage: H=PermutationGroup([a,b])   # could have used H=S10.subgroup([a,b]) instead
sage: H.order()
120
sage: a*b*a*b^2
(1,4,3,2)
sage: S10("(1,4,3,2)") in H
true
sage: S10("(8,9,10)") in H
false
```

(5) Some subgroups of the dihedral group $D_4 = \{R_0, R_{90}, R_{180}, R_{270}, H, V, D, D'\}$ are
$\langle R_{90}\rangle = \{R_0, R_{90}, R_{180}, R_{270}\}$
$\langle R_{180}\rangle = \{R_0, R_{180}\}$
$\langle V\rangle = \{R_0, V\}$
$\langle H, V\rangle = \{R_0, R_{180}, H, V\}$

If we construct only the portion of the multiplication table that involves $\{R_0, R_{180}, H, V\}$ then we can immediately see that it is a subgroup since it is closed under the operation, and inverses.

Sage

```
sage: D4=DihedralGroup(4)
sage: D4sublist=["()","(1,3)(2,4)", "(1,4)(2,3)", "(1,2)(3,4)"]
sage: D4subnames=["R0","R180","H","V"]
sage: D4.cayley_table(names=D4names,elements=D4list)
 *    R0 R180    H    V
    +---------------------
  R0|   R0 R180    H    V
R180| R180   R0    V    H
   H|    H    V   R0 R180
   V|    V    H R180   R0
```

11.3 The Center of a Group

The **center** of a group G is the subset $Z(G)$ of all elements that commute with every element of G:

$$Z(G) = \{a \in G \mid ag = ga \text{ for all } g \in G\}.$$

Theorem 11.2: *For a group G the center $Z(G)$ is a subgroup of G.*

Proof: The identity is in $Z(G)$, therefore $Z(G) \neq \emptyset$. If a and b are in $Z(G)$ then for any $g \in G$, $(ab)g = a(bg) = a(gb) = (ag)b = (ga)b = g(ab)$ so $ab \in G$. Also, $ag = ga$ implies $ga^{-1} = a^{-1}g$ so $a^{-1} \in G$. Therefore, by the Two-Step subgroup test $Z(G) < G$. □

Note that $Z(G) = G$ if and only if G is abelian. We've shown in Lecture 3 Exercise 9 that for every non-trivial permutation in S_n, where $n \geq 3$, there exists one that does not commute with it. This means $Z(S_n) = \{\varepsilon\}$. However, for subgroups of S_n this is not necessarily the case. For example A_3 is abelian and so $Z(A_3) = A_3$. Here we verify this in Sage.

Sage

```
sage: A3=AlternatingGroup(3)
sage: A3.center()
Permutation Group with generators [(1,2,3)]
sage: A3.center().list()
[(), (1,2,3), (1,3,2)]
```

In Section 21.3.1 we determine the center of the Rubik's Cube group.

11.4 Lagrange's Theorem

Looking back at our examples in last section we make the following observation: *the order of a subgroup divides the order of the group*. For example, in S_3, which is a group of order 6, all the subgroups we listed are either of order 1, 2 or 3, which are precisely the divisors of 6. Verify this observation for the other examples.

This raises the question: Must the order of a subgroup always be a divisor of the order of the group? If this is true, then it puts a pretty strict condition on the possible subsets that can be subgroups. For instance, we would be able to quickly conclude that $\{R_0, H, D\}$ is not a subgroup of D_4 since 3 does not divide 8.

It turns out that our observation is true in general. This is known as Lagrange's Theorem.

Theorem 11.3 (Lagrange's Theorem): *If G is a finite group and H is a subgroup of G, then $|H|$ divides $|G|$.*

We will prove this theorem in Section 18.2 of Lecture 18. Also see Exercise 15 in Lecture 17.

If we consider the subgroup $\langle g \rangle$ generated by an element $g \in G$, then the order of this subgroup is precisely the order of g. In other words, our two definitions of the word "order" (both as the size of a group, and the smallest number n for which $g^n = e$) agree.

Corollary 11.4 ($ord(a)$ divides $|G|$): *In a finite group, the order of each element divides the order of the group.*

Here is some experimental evidence in support of the corollary.

Sage

```
sage: n=20
sage: Zn = CyclicPermutationGroup(n)
sage: element_orders=Set([g.order() for g in Zn])
sage: element_orders
{1, 2, 4, 5, 10, 20}
```

The orders of the elements in $\mathbb{Z}_{20}$ are all divisors of 20. We could vary n, try dihedral groups or unit integer groups, etc. In every case, we would find that the order of an element must divide the order of the group.

As a partial converse to Corollary 11.4 we have the following.

Theorem 11.5 (Cauchy's Theorem): *Let p be a prime dividing $|G|$. Then there is a $g \in G$ of order p.*

Note that for non-prime divisors d of $|G|$ it is not true in general that G contains an element of order d. For example, A_4 is a group of order 12 but it does not contain an element of order 4 (Why?).

A proof of Cauchy's Theorem will be deferred to a later lecture (if we have time).

11.5 Cyclic Groups Revisited

In a cyclic group $G = \langle g \rangle$ every element is of the form g^k for some k. If G is infinite then every distinct power of g is a distinct element of G. Think about $\mathbb{Z} = \langle 1 \rangle$ under addition as an example (of course, here we have to reinterpret "power" to mean "multiple" since the group operation is addition),

If $G = \langle g \rangle$ is finite of order n then $G = \{e, g, g^2, \dots, g^{n-1}\}$ and $g^i = g^j$ if and only if $n \mid j - i$.

This means it is fairly easy to work with cyclic groups, since taking products and determining when two elements are really the same, is a fairly simple task.

The following theorems list some nice properties that cyclic groups have, including how to find all subgroups and all elements of a particular order.

Theorem 11.6 (Fundamental Theorem of Cyclic Groups): *Every subgroup of a cyclic group is cyclic. Moreover, if $|\langle g \rangle| = n$ then for each divisor of k of n there is exactly one subgroup of $\langle g \rangle$ of order k.*

The proof of this is well within our reach, but I will not prove it here.

Theorem 11.7 (Generators of Cycle Groups): *Let $G = \langle g \rangle$ be a cyclic group of order n. Then $G = \langle g^k \rangle$ if and only if $\gcd(k, n) = 1$.*

Theorem 11.8 (Number of elements of each order in a cyclic group.): *If d is a divisor of n, the number of elements of order d in a cyclic group of order n is $\phi(d)$.*

In the specific case when the group is $\mathbb{Z}_n$, and the operation is addition, these theorems can be restated as follows.

Theorem 11.9 (Generators, Subgroups, and Orders in $\mathbb{Z}_n$): *Consider the group of integers modulo n, $\mathbb{Z}_n$.*

(a) An integer k is a generator of $\mathbb{Z}_n$ if and only if $\gcd(k, n) = 1$.

(b) For each divisor k of n, the set $\langle n/k \rangle$ is the unique subgroup of $\mathbb{Z}_n$ of order k, moreover, these are the only subgroups of $\mathbb{Z}_n$.

(c) For each $k \mid n$ the elements of order k are of the form $\ell \cdot (n/k)$ where $\gcd(\ell, k) = 1$. The number of such element is $\phi(k)$, and each of these is a generator of the unique subgroup of order k.

Example 11.2: Let's determine all the subgroups of $\mathbb{Z}_{24}$. By Theorem 11.9 the generators of $\mathbb{Z}_{24}$ are precisely the elements which are relatively prime to $24 = 2^3 3$. These are $1, 5, 7, 11, 13, 17, 19, 23$.

$\langle 1 \rangle = \mathbb{Z}_{24}$

2 is an element of order 12, so it generates a cyclic subgroup of order 12:

$\langle 2 \rangle = \{0, 2, 3, 6, 8, 10, 12, 14, 16, 18, 20, 22\}$.

The other generators are $k \cdot 2$ where k is relatively prime to 12. Since there are $\phi(12) = 4$ numbers relatively prime to 12, namely $\{1, 5, 7, 11\}$ then the other generators of this subgroup are $5 \cdot 2 = 10$, $7 \cdot 2 = 14$, $11 \cdot 2 = 22$.

3 is an element of order 8, so it generates a cycle subgroup of order 8:

$\langle 3 \rangle = \{0, 3, 6, 9, 12, 15, 18, 21\}$.

Other generators of this subgroup are $m \cdot 3$ where m is relatively prime to 8. There are $\phi(8) = 4$ such generators: $3, 9, 15, 21$.

We can continue looking for subgroups (and generators) in this way. We just keep in mind that to find a subgroup of size k we look for an element of order k, since it will generate the only subgroup of size k. This is what Theorem 11.9 (and more generally Theorem 11.6) states.

Table 11.1 lists all subgroups, orders and generators of $\mathbb{Z}_{24}$.

subgroup	**order**	**other generators**
$\langle 1 \rangle = \mathbb{Z}_{24}$	24	$5, 7, 11, 13, 17, 19, 23$
$\langle 2 \rangle = \{0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22\}$	12	$10, 14, 22$
$\langle 3 \rangle = \{0, 3, 6, 9, 12, 15, 18, 21\}$	8	$9, 15, 21$
$\langle 4 \rangle = \{0, 4, 8, 12, 16, 20\}$	6	20
$\langle 6 \rangle = \{0, 6, 12, 18\}$	4	18
$\langle 8 \rangle = \{0, 8, 16\}$	3	16
$\langle 12 \rangle = \{0, 12\}$	2	
$\langle 0 \rangle = \{0\}$	1	

Table 11.1: Subgroups of $\mathbb{Z}_{24}$

11.6 Cayley's Theorem

We've mostly been focussing our attention on permutation groups. One may wonder whether we are limiting ourselves and missing out on some pretty important groups that we wouldn't otherwise see. Well, as it turns out, *every* group is really just a permutation group, the difference is only in how we name the elements. To see this, let G be a finite group of order n. List the elements of G:

$$g_1, \quad g_2, \quad g_3, \quad \cdots \quad , g_{n-1} \quad , g_n.$$

For any element $a \in G$ multiply the elements of the list by a:

$$ag_1, \quad ag_2, \quad ag_3, \quad \ldots, \quad ag_{n-1}, \quad ag_n.$$

This is just a permutation of the list of elements in G. Why? In other words, we can associate to the element a the permutation that it induces on the elements of G. It turns out that the set of all such permutations contains all the information about G. In other words, we can just think of G as a set of permutations. This is known as Cayley's theorem.

Theorem 11.10 (Cayley's Theorem): *Let G be a group. For each $a \in G$, define a mapping*

$$\begin{aligned}\rho_a : \ & G \to G \\ & x \mapsto ax.\end{aligned}$$

Then

(a) ρ_a is a permutation of the set G,

(b) $H = \{\rho_a \mid a \in G\}$ is a subgroup of S_G, the group of all permutations of the set G.

(c) H and G are essentially the same groups, all that is different is the names of the elements. More precisely, $ab = c$ in G if and only if $\rho_a\rho_b = \rho_c$ as permutations.

We only note the theorem here since it tells us that we aren't, in a sense, limiting ourselves by studying only permutation groups. Also, this theorem indicates why Sage uses permutation groups to represent other groups.

11.7 Exercises

1. Is $\{\varepsilon, (1,2), (1,2,3)\}$ a subgroup of S_4?
2. Name the elements in S_3 as follows:

$$s_1 = (1,2), \quad s_2 = (1,3), \quad s_3 = (2,3), \quad s_4 = (1,2,3), \quad s_5 = (1,3,2).$$

   (a) Let G be the subgroup generated by s_1, $G = \langle s_1 \rangle$. Verify there are only two elements in G.
   (b) What is the order of s_4?
   (c) Let H be the permutation group with generator s_5, $G = \langle s_5 \rangle$. Verify that there are only three elements in H.
   (d) Show that $S_3 = \langle (1,2), (1,3,2) \rangle$. In other words, show that S_3 is generated by $(1,2)$ and $(1,3,2)$.

3. Let $G = \langle (1,2), (3,4,5) \rangle$. Show that G is a subgroup of S_5 of order 6.
4. Find a subgroup of order 4 in S_4.

5. Find a subgroup of order 8 in S_4.

6. (a) List all the elements of A_4.
   (b) List all the subgroups of A_4.
   (c) Show that the converse of Lagrange's Theorem is false by finding a divisor of $|A_4|$ for which there is no subgroup of that order.

Dihedral Groups:

7. Determine all the subgroups of D_3.

8. Find the center $Z(D_4)$ of D_4.

9. Determine all the subgroups of D_5.

10. (a) Determine the number of elements of order 2 in D_n.
    (Hint: You will need to consider separately the cases when n is even and n is odd.)
    (b) How many subgroups of order 2 does D_n have?

11. Determine the orders of the elements in D_{33} and how many there are of each.

12. How many elements of order 4 does D_{12} have? How many elements of order 4 does D_{4n} have?

13. Let n be an odd integer. Prove that every subgroup of D_n of odd order is cyclic.

Group of Integers under addition modulo n:

14. Find all the subgroups, and determine generators for each subgroup, for each of the following.

    (a) $\mathbb{Z}_8$ (b) $\mathbb{Z}_{12}$ (c) $\mathbb{Z}_{17}$

15. Find all the elements of order 6 in $\mathbb{Z}_{18}$.

16. Find all the elements of order 15 in $\mathbb{Z}_{30}$.

17. Find all the elements of order 10 in $\mathbb{Z}_{40}$.

18. List all the elements of order 8 in $\mathbb{Z}_{8000000}$.

Unit Group modulo n:

19. Determine all the subgroups of $U(12)$.

20. For each value of n listed below, determine whether or not $U(n)$ is a cyclic group. When it is cyclic, list all of the generators of $U(n)$, $n = 5, 9, 10, 14, 15, 18, 20, 22, 25$. Make a conjecture about the prime power decomposition of integers n for which $U(n)$ is cyclic. Are $n = 9$ and $n = 16$ counterexamples of your conjecture? (Try them.) If so, modify your conjecture.

21. Given the fact that $U(49)$ has 42 elements, determine the number of generators that $U(49)$ has without actually finding any of the generators.

22. Prove that $U(2^n)$ $(n \geq 3)$ is not cyclic.
    (Hint: Look for a property that $U(2^n)$ has but cyclic groups do not have.)

Subgroups in General:

23. Prove that a group of order 3 must be cyclic.

24. Suppose that G is a cyclic group for which 6 divides $|G|$. How many elements of order 6 does G have? If 8 divides $|G|$, how many elements of order 8 does G have? If a is one element of order 8, list the other elements of order 8.

25. Let $|G| = 33$. What are the possible orders for the elements of G? Show that G must have an element of order 3.

26. Let $|G| = 8$. Show that G must have an element of order 2. Show by counterexample that G need not have an element of order 4.

27. If G is an abelian group and contains cyclic subgroups of orders 4 and 5, what other sizes of cyclic subgroups must G contain.

28. If G is an abelian group and contains a pair of subgroups of order 2, show that G must contain a subgroup of order 4. Must this subgroup be cyclic?

29. Show that every group of order at most 4 is abelian. This says that groups of order ≤ 4 don't have enough room to have elements that don't commute.

30. Show that if G is a group where $|G| = p$ is prime then G is cyclic.

31. Let G be a group such that $|G| = p^n$, where p is prime. Show that G has an element of order p.

32. Let G be a group such that $|G| = p^2$. Show that either G is cyclic, or $a^p = e$ for all $a \in G$.

33. **One-Step Subgroup Test**. Let G be a group and H a nonempty subset of G. Show that H is a subgroup of G if $ab^{-1} \in H$ for every $a, b \in H$.

34. **Finite Subgroup Test** Let G be a finite group and H a nonempty subset of G. Show that H is a subgroup of G if H is closed under multiplication.

Lecture 12

Puzzle Groups

In this lecture we associate a group to each permutation puzzle, called the *puzzle group*. We'll see that this group can be represented by a group of permutations, allowing us to use Sage to investigate the puzzles.

12.1 Puzzle Groups

Let's first recall the definition of a *permutation puzzle*, since we would like to see how groups come into the picture. In Lecture 1 we defined a *one person game*, and from that we gave the following definition of a permutation puzzle.

A **permutation puzzle** is a one person game (solitaire) with a finite set $T = \{1, 2, \ldots, n\}$ of puzzle pieces satisfying the following four properties:

1. For some $n > 1$ depending only on the puzzle's construction, each move of the puzzle corresponds to a unique permutation of the numbers in T,
2. If the permutation of T in (1) corresponds to more than one puzzle move then the two positions reached by those two respective moves must be indistinguishable,
3. Each move, say M, must be "invertible" in the sense that there must exist another move, say M^{-1}, which restores the puzzle to the position it was at before M was performed, In this sense, we must be able to "undo" moves.
4. If M_1 is a move corresponding to a permutation τ_1 of T and if M_2 is a move corresponding to a permutation τ_2 of T then $M_1 \cdot M_2$ (the move M_1 followed by the move M_2) is either
   - not a legal move, or
   - corresponds to the permutation $\tau_1\tau_2$.

As indicated in part 4 it may happen that the composition of two moves is not legal. For example, this happens with the 15-Puzzle since legal moves change as the empty space moves around the board. This generally happens when dealing with a puzzle that contains a "gap". We won't consider such puzzles in this lecture, besides a remark in Section 12.4. Instead we will focus on puzzles for which two moves can always be composed. Typically these are the puzzles "without-gaps".

Let Puz be a permutation puzzle (where any two moves can be composed). For example Puz could be Rubik's cube, Oval Track, or Hungarian Rings. We consider two puzzle moves, m_1 and m_2, to be *equivalent* if the two positions reached by those two respective moves are indistinguishable.

Let $M(\text{Puz})$ be the set of all inequivalent puzzle moves (what we typically refer to as move-sequences). We can think of $M(\text{Puz})$ as just the set of all possible configurations, or positions (permutations) of the puzzle pieces. We have a way to combine elements of $M(\text{Puz})$: if $m_1, m_2 \in M$ then $m_1 m_2$ represents the move-sequence m_1 followed by m_2, which is again in $M(\text{Puz})$. (This is why we assume the puzzle does not have gaps.) It turns out that $M(\text{Puz})$ is a group under this operation. The identity is the "do nothing" move, and inverses exist by part 3 of the definition above. Associativity follows from the fact that "moves" correspond to "permutations" and permutation composition is associative.

Definition 12.1 (Puzzle Group): For a permutation puzzle Puz, the set of all inequivalent puzzle moves $M(\text{Puz})$ is a group under move composition. $M(\text{Puz})$ is called the **puzzle group** of Puz.

Since puzzle moves and positions correspond to permutations we can represent $M(\text{Puz})$ as a subgroup of a permutation group. To do this we just need to associate each basic legal move $m_i \in M(\text{Puz})$, $1 \leq i \leq k$, to a permutation α_i. We then use the permutation group $\langle \alpha_1, \alpha_2, \ldots, \alpha_k \rangle$ to represent the puzzle. We've already done this with all of our puzzles, so here we are just emphasizing the fit within group theory.

12.2 Rubik's Cube

Let Puz be an $n \times n \times n$ Rubik's cube, then we call $M(\text{Puz})$ the **n-cube group**. In the special case when $n = 3$ we call it the **Rubik's cube group**. We use the special notation RC_n to denote the n-cube group.

12.2.1 3-Cube Group

As we described in Lecture 1 label the facets of the Rubik's Cube as shown Figure 12.1. Figure 12.2 shows the labeling on a 3-dimensional cube.

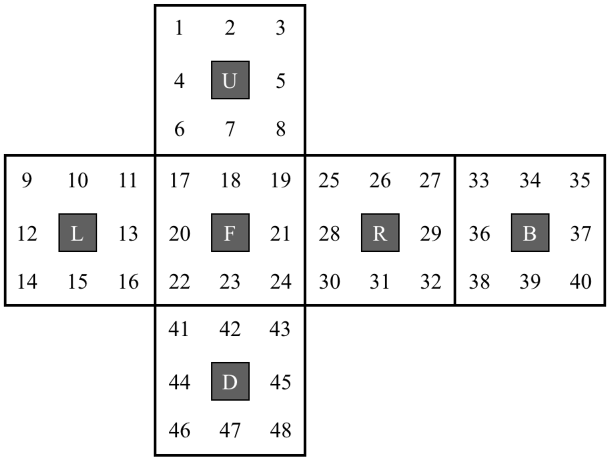
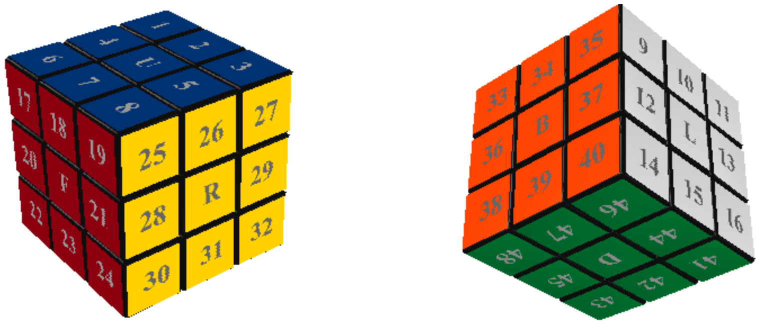


Figure 12.1: Facet labeling on the Rubik's cube.



(a) Labeling on Up, Right, Front faces

(b) Labeling on Down, Back, Left faces

Figure 12.2: The labeling of the facets of Rubik's Cube.

The permutation corresponding to each of the basic moves of the Rubik's Cube are:

$$\begin{aligned}
R &= (25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)\\
L &= (9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)\\
U &= (1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)\\
D &= (41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)\\
F &= (17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)\\
B &= (33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)
\end{aligned}$$

$R^{-1}, L^{-1}, U^{-1}, D^{-1}, F^{-1}, B^{-1}$ correspond to the inverses of these permutations.

Since the centre's of the cube are fixed by these moves then any two of these moves are inequivalent. This means that RC_3 can be represented by the subgroup of S_{48} generated by these permutations:

$$RC_3 = \langle \mathrm{R}, \mathrm{L}, \mathrm{U}, \mathrm{D}, \mathrm{F}, \mathrm{B} \rangle.$$

We can define RC_3 in Sage as follows.

Sage

```
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
sage: RC3=S48.subgroup([R,L,U,D,F,B])      # define Rubik's cube group to be RC3
```

Now that RC_3 is in Sage we can calculate some facts about the Rubik's cube. For example, we can determine the size of RC_3. This is the number of different configurations there are of the cube.

Sage

```
sage: RC3.order()
43252003274489856000
sage: factor(RC3.order())
2^27 * 3^14 * 5^3 * 7^2 * 11
```

Therefore there are approximately $4.3 \cdot 10^{19}$ configurations of the cube. And only one solution!

Theorem 12.1: *The Rubik's cube group RC_3 has order* $2^{27}3^{14}5^{3}7^{2}11 = 43,252,003,274,489,856,000$.

Since the order of an element in a group must divide the size of the group, then we immediately see from the factored form of $|RC_3|$ that there are no elements of prime order ≥ 13. Also, by Cauchy's theorem (see Lecture 11), there must be an element of order 11. Actually finding such an element is another story, all we know is one exists. In fact, 9 others must exist as well since it would generate a subgroup of order 11.

We can also check if it is possible to flip a single edge, while leaving everything else in place. Consider flipping the cubie in the *uf* cubical, the corresponding permutation is $(7,18)$. The following calculation shows it is not in RC_3.

Sage

```
sage: S48("(7,18)") in RC3
False
```

However, we can flip two edges, say for example the cubies in the *uf* and *ur* cubicals. This corresponds to the permutation $(7,18)(5,26)$.

Sage

```
sage: S48("(7,18)(5,26)") in RC3
True
```

Notice this only tells us that it is possible to flip two edges using moves R, L, U, D, F, B, but it doesn't indicate what sequence of moves will do this. This is in fact a much harder problem. Basically what we are asking for is a method which can determine, for any element of RC_3, a way to write it as a product of the generators (or equivalently, as a word in R, L, U, D, F, B). This is known as the *word problem* in group theory and is very difficult in many situations.

However, Sage does contain an implementation of an algorithm for solving the word problem in RC_3. It doesn't necessarily return the shortest possible move sequence, but it does a pretty good job nonetheless. For this we need to use the built-in `CubeGroup()` package.

Sage

```
sage: rubik=CubeGroup();
sage: G=rubik.group();
sage: B=rubik.B();
sage: state = G("(7,18)(5,26)")
sage: rubik.solve(state)     # calls the solve algorithm
"F2 R2 B′ F′ D′ F D B R2 F′ R′ F′ R"
```

Therefore, one move-sequence for flipping edges uf and ur is

$$F^2R^2B^{-1}F^{-1}D^{-1}FDBR^2F^{-1}R^{-1}F^{-1}R.$$

12.2.2 2-Cube Group

Label the facets of the Pocket Cube as shown in Figure 12.3 and Figure 12.9.

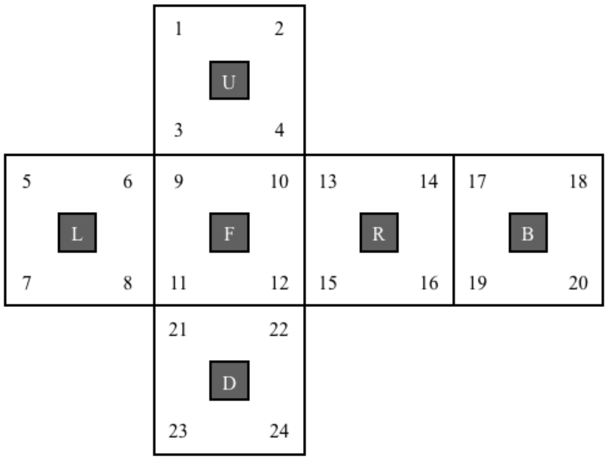
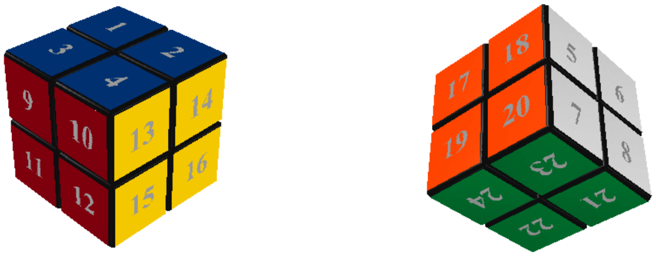


Figure 12.3: Facet labeling on the Pocket cube.



(a) Labeling on Up (blue), Right (yellow), Front (red) faces

(b) Labeling on Down (green), Back (orange), Left (white) faces

Figure 12.4: The labeling of the facets of the Pocket Cube.

The permutation corresponding to each of the basic moves of the Pocket Cube are:

$$
\begin{aligned}
R &= (13,14,16,15)(10,2,19,22)(12,4,17,24)\\
L &= (5,6,8,7)(3,11,23,18)(1,9,21,20)\\
U &= (1,2,4,3)(9,5,17,13)(10,6,18,14)\\
D &= (21,22,24,23)(11,15,19,7)(12,16,20,8)\\
F &= (9,10,12,11)(3,13,22,8)(4,15,21,6)\\
B &= (17,18,20,19)(1,7,24,14)(2,5,23,16)
\end{aligned}
$$

$R^{-1}, L^{-1}, U^{-1}, D^{-1}, F^{-1}, B^{-1}$ correspond to the inverses of these permutations.

There is one major difference between the Pocket cube and Rubik's cube: the Pocket cube does not have any fixed centres. Why does this matter? Consider the moves R and L. They are *equivalent*! Notice that applying R, leaves the cube in exactly the same position as L (the cube as a whole has just been rotated in space). Another way to say this is RL^{-1} is the identity in RC_2. Try it!

But if we were to use the permutations above to generate a group then this wouldn't be the group RC_2. Since the product of permutations associated with R and L don't have the property that $RL^{-1} = \varepsilon$. This means the permutations are picking up the fact that the cube rotated in space.

Again let's summarize the real difference between Rubik's cube and the Pocket cube: the Pocket cube can be rotated in space using only puzzle moves (which rotate faces), whereas Rubik's cube cannot be rotated in space using puzzle moves (since centres stay fixed under face rotations).

This means that RC_2 is smaller that the permutation group generated by the 6 permutations above. In fact, we really only need one of each of the following pairs of moves: $\{R, L\}, \{U, D\}, \{F, B\}$. We'll choose to only use R, D, F. This means the UBL cubie always remains in its home position. This is the piece we will keep fixed.

Sage

```
sage: S24=SymmetricGroup(24)
sage: R=S24("(13,14,16,15)(10,2,19,22)(12,4,17,24)")
sage: D=S24("(21,22,24,23)(11,15,19,7)(12,16,20,8)")
sage: F=S24("(9,10,12,11)(3,13,22,8)(4,15,21,6)")
sage: RC2=S24.subgroup([R,D,F])      # define Pocket cube group to be RC2
```

We can determine the size of RC_2.

Sage

```
sage: RC2.order()
3674160
sage: factor(RC2.order())
2^4 * 3^8 * 5 * 7
```

Therefore there are approximately 3.6 million configurations of the Pocket cube. And only one solution.

Theorem 12.2: *The Pocket cube group RC_2 has order $2^4 3^8 5 \cdot 7 = 3,674,160$.*

If we didn't realize that some moves are equivalent, and just constructed the group generated by all moves, what would happen?

Sage

```
sage: S24=SymmetricGroup(24)
sage: R=S24("(13,14,16,15)(10,2,19,22)(12,4,17,24)")
sage: L=S24("(5,6,8,7)(3,11,23,18)(1,9,21,20)")
sage: U=S24("(1,2,4,3)(9,5,17,13)(10,6,18,14)")
sage: D=S24("(21,22,24,23)(11,15,19,7)(12,16,20,8)")
sage: F=S24("(9,10,12,11)(3,13,22,8)(4,15,21,6)")
sage: B=S24("(17,18,20,19)(1,7,24,14)(2,5,23,16)")
sage: S24.subgroup([R,L,U,D,F,B]).order()
88179840
sage: 88179840/3674160
24
```

We would have been off by a factor of 24. Why 24? This is precisely the number of different rotations there are for the whole cube. Since the permutation group was treating rotations of the cube as different states, but the cube group RC_2 should know these states really aren't different at all, then it is no surprise that we would be off by the number of rotations to the cube: 24.

This does illustrate, however, that we can't just assign a permutation to each move, and form the permutation group. Some thought needs to be taken as to whether the representation is faithful.

Swapping Corners on the Pocket Cube:

Are we able to swap two corners on the Pocket Cube, while keeping every other cubie in its home location (not necessarily with proper orientation)?

Think about what a typical permutation would look like. Since corners can possibly be twisted when returned to their home locations, it is not simply a matter of asking if a 2-cycle is in RC_2. However, we aren't really interested in how the stickers move around, just the cubies themselves. If we view RC_2 acting on the the 8 cubies, we just want to know if we can swap two cubies, and fix all other cubies in their current location.

If we number the cubicles as follows: 1 is the *ufr* cubical, 2 is the *urb* cubical, 3 is the *ubl* cubical, 4 is the *ulf* cubical, 5 is the *dfr* cubical, 6 is the *drb* cubical, 7 is the *dbl* cubical, 8 is the *dlf* cubical.

The action of each move on the cubies are then:

$$\begin{aligned} R &= (1,2,6,5) \\ L &= (3,4,8,7) \\ U &= (1,4,3,2) \\ D &= (5,6,7,8) \\ F &= (1,5,8,4) \\ B &= (2,3,7,6) \end{aligned}$$

We can the ask Sage to compute whether it is possible to swap the 1 and 2 cubies.

Sage

```
sage: S8=SymmetricGroup(8)
sage: R=S8("(1,2,6,5)")
sage: L=S8("(3,4,8,7)")
sage: U=S8("(1,4,3,2)")
sage: D=S8("(5,6,7,8)")
sage: F=S8("(1,5,8,4)")
sage: B=S8("(2,3,7,6)")
sage: H=S8.subgroup([R,L,U,D,F,B])
sage: S8("(1,2)") in H
True
sage: H.order()=factorial(8)
True
```

The computation shows we can not only swap cubies 1 and 2 but in fact every permutation of the 8 cubies is possible. Remember though, the representation of RC_2 that we chose to work with here ignores any twisting of corners. So even though we can move the pieces anywhere we want, there may be limitations on how we can twist them. We will investigate how the corners can twist in Lecture 20.

12.2.3 Oval Track

Let Puz be the Oval Track puzzle (or one of its variations), then we call $M(\mathrm{Puz})$ the **Oval Track group** and we use the notation OT to denote this group.

We'll look at a few different variations of the puzzle, corresponding to different modifications of the turntable move T.

12.2.4 Oval Track - TopSpin: $T = (1,4)(2,3)$

The basic legal moves of the TopSpin version of the Oval Track puzzle are R, and T, where R denotes a clockwise rotation of numbers around the track, where each number moves one space, and T denotes a rotation of the turntable. See Figure 12.5.

The permutation corresponding to the legal moves R, and T are as follows:

$$\begin{aligned} R &= (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20) \\ T &= (1,4)(2,3) \end{aligned}$$

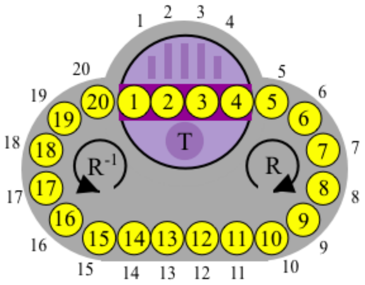


Figure 12.5: The Oval Track Puzzle.

Note that $T^{-1} = T$. This is due to the fact that spinning the turntable in either direction achieves the same result.

The basic moves R and T are not equivalent, so OT can be represented by the permutation group generated by these two permutations.

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4)(2,3)")
sage: OT=S20.subgroup([R,T])        # define OT to be a permutation group
```

What is the size of OT? The puzzle consists of permuting 20 disks, so it is natural to wonder if all permutations are possible. Since there are $20!$ permutations of 20 objects, we'd like to know if $|OT| = 20!$.

Sage

```
sage: OT.order()==factorial(20)
True
```

This means OT is actually the symmetric group of degree 20: $OT = S_{20}$. Therefore, every permutation of the disks is possible. Of course, the key to solving this puzzle is to figure out how you can obtain each permutation using only moves R and T.

12.2.5 Oval Track - Variation 2: $T = (1, 4, 3, 2)$

The *turntable move* in the original TopSpin puzzle is now replaced with the move indicated by the purple dashed lines. In this version, the new *turntable move* for the puzzle in Figure 12.6 moves the disk in spot 4 to spot 3, the disk in spot 3 to spot 2, the disk in spot 2 to spot 1, and takes the disk in spot 1 to spot 4.

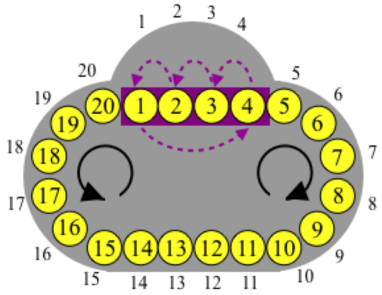


Figure 12.6: The Oval Track Puzzle.

The permutation corresponding to the legal moves R, and T are as follows:

$$R = (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)$$
$$T = (1,4,3,2)$$

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4,3,2)")
sage: OT2=S20.subgroup([R,T])        # define OT2 to be a permutation group
sage: OT2.order()==factorial(20)
True
```

In this variation all possible permutations of the 20 disks are possible.

12.2.6 Oval Track - Variation 3: $T = (1,6)(2,5)(3,4)$

Another version of the *turntable move* involving 6 disks is given in Figure 12.7.

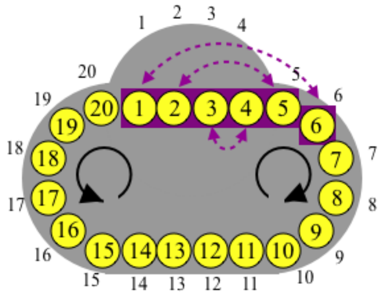


Figure 12.7: The Oval Track Puzzle.

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,6)(2,5)(3,4)")
sage: OT3=S20.subgroup([R,T])        # define OT3 to be a permutation group
sage: OT3.order()==factorial(20)
True
```

In this variation all possible permutations of the 20 disks are possible.

12.3 Hungarian Rings

Let Puz be the Hungarian Rings puzzle (numbered version), then we call $M(\mathrm{Puz})$ the **Hungarian Rings group** and we use the notation HR to denote this group.

The basic legal moves of the Hungarian Rings puzzle are R, and L, where R denotes a clockwise rotation of numbers around the right-hand ring (each number moves one space), and L denotes a clockwise rotation of numbers around the left-hand ring.

The permutation corresponding to each of the legal moves R and L are:

$$R = (1,38,37,36,35,6,34,33,32,31,30,29,28,27,26,25,24,23,22,21)$$
$$L = (1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)$$

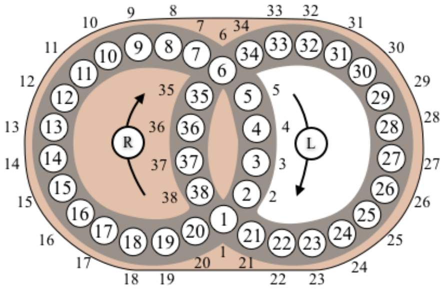


Figure 12.8: Hungarian Rings - numbered version.

R^{-1} and L^{-1} correspond to the inverses of these permutations.

Since the moves R and L are inequivalent then HR can be represented by the group of permutations generated by R and L.

Sage

```
sage: S38=SymmetricGroup(38)
sage: L=S38("(1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)")
sage: R=S38("(1,38,37,36,35,6,34,33,32,31,30,29,28,27,26,25,24,23,22,21)")
sage: HR=S38.subgroup([R,L])==factorial(38)
True
```

Therefore, all possible permutations of the 38 balls are possible.

12.4 15-Puzzle

The 15-puzzle does not fit into group theory as neatly as our other puzzles do. The problem is that a move must involve the empty space, so the available legal moves at each stage changes depending on where the empty space is.

In what follows, we will describe a move of the pieces of the 15 puzzle by the first letter of the word *(u)p*, *(d)own*, *(l)eft*, *(r)ght*, which is to indicate the direction a tile is pushed into the empty space. For example, beginning with the empty space in spot 16, let m_1 be the sequence of moves:

$$m_1 = rrr.$$

Similarly, with the empty space in spot 16, let m_2 be the sequence of moves:

$$m_2 = rddd.$$

Move m_1 places the empty space in spot 13 by moving all tile on the bottom row to the right. Whereas, move m_2 places the empty space in spot 3. Therefore, it is impossible to perform the move sequence $m_1m_2 = (rrr)(rddd)$ since once three r moves are applied there is no tile to the left of the empty space to apply another r move. The set of all legal moves is not closed under composition, therefore is not a group.

However, if we narrow our focus we can find a group lurking in there somewhere.

Represent each sequence of moves by its corresponding permutation, so the set of all such move sequence corresponds to a subset of the permutation group S_{16}. Let this subset be denoted by FP:

$$FP = \{\alpha \mid \alpha \text{ is the permutation corresponding to a legal position of the 15-puzzle}\}.$$

We already noted FP is not a group but the example gives us some insight into how we can fix this. If each moves starts with the empty space in box 16, then returns it to box 16, then the next move can be applied without any trouble. We let FP^* consist of the set of all moves that leaves the empty space in spot 16. In terms of permutations this means:

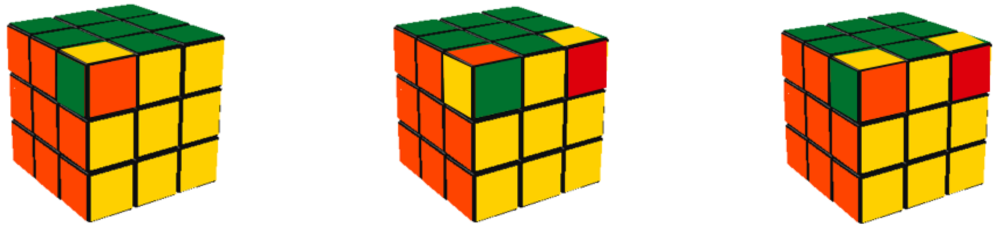
$$FP^* = \{\alpha \in FP \mid \alpha(16) = 16\}.$$

Now FP^* is a group. In fact we know it to be the group A_{15}.

In general when considering puzzles with gaps, we can look at the subset of legal moves where each move returns the space to its home position, this set will form a group.

12.5 Exercises

1. **Single Corner Twist.** Is it possible to rotate a single corner cubie of Rubik's cube, while leaving everything else in its home position? See Figure 12.9a.



(a) Figure for Exercise 1

(b) Figure for Exercise 2

(c) Figure for Exercise 3

Figure 12.9: Which corner twists are possible?

2. **Two Corner Twists.** For Rubik's cube, is it possible to rotate two corner cubies in the same direction, while leaving everything else in its home position? See Figure 12.9b.

3. **Another Two Corner Twists.** For Rubik's cube, is it possible to rotate two corner cubies in opposite directions, while leaving everything else in its home position? See Figure 12.9c.

4. **Swapping Corners on Rubik's cube.** Show that it is impossible to swap two corner cubies on Rubik's cube, while leaving all other cubies in their home locations (not necessarily with proper orientation)?

5. **Oval Track with 19 Disks.** Consider the Oval Track puzzle (TopSpin version) where only 19 disks are used. Are all permutations of the 19 disks possible? If not, can you describe exactly which permutations are possible?

6. **Varying the Number of Disks on Oval Track.** For the Oval Track puzzle with n disk, let OT_n denote the puzzle group, determine the size of OT_n, for $6 \leq n \leq 20$. In each case, describe exactly which permutations of the puzzle pieces are possible.

7. **Very Few Disks on Oval Track.** Consider OT_n for $n = 4, 5$. Investigate which permutations of the puzzle pieces are possible.

8. **Varying the turntable move T of the Oval Track puzzle.** In this exercise you will investigate, with the help of Sage, some variations of the Oval Track puzzle. In all variations[1], we assume there

[1] Variation names are due to John O. Kiltinen who studies these in his book: *Oval Track and other Permutation Puzzles.*

are 20 disks, and the usual move consisting of rotating the pieces along the track is R. We will vary the turntable move T. We have already seen that if the turntable move is $T = (1,4)(2,3)$ or $T = (4,3,2,1)$ then we are still able to obtain *all* permutations of the 20 disks. Investigate the other variations of the move T given in Table 12.1. Under the column "permutation group", try to determine what group of permutations of the 20 pieces is possible. The first two rows have been filled in already.

variation	**turntable move T**	**permutation group**
OT 1	$(1,4)(2,3)$	S_{20}
OT 2	$(4,3,2,1)$	S_{20}
OT 3	$(3,2,1)$	
OT 4	$(5,4,3,2,1)$	
OT 5	$(1,2)(3,4)$	
OT 6	$(1,11)(4,14)$	
OT 7	$(5,3,1)$	
OT 8	$(1,3)(2,4)$	

Table 12.1: Variations of the Oval Track puzzle

Lecture 13

Commutators

> *fanwuq:* Just solve one corner at a time like LBL until you get to last layer. Then, you can just use commutators to solve the rest of the corners.
>
> *JBogwith:* I'm sorry, I don't understand. I can get to the last layer, it is then where I get stuck. **What are commutators?**
>
> *www.speedsolving.com forum discussion. Dec. 2007*

In this lecture we look at a product known as a *commutator*. This type of move sequence is useful for *creating* moves on permutation puzzles.

13.1 Commutators

When playing with permutation puzzles, certain move sequences can occur more often than others. For instance, a move sequence of the form "move 1, then move 2, then inverse of move 1, then inverse of move 2" turns out to be quite useful. This type of move is called a "commutator". As you read through this lecture, you will find it useful to have a puzzle in-hand to try things out for yourself.

Definition 13.1: If g, h are two elements of a group G, then we call the element

$$[g, h] = ghg^{-1}h^{-1}$$

the **commutator** of g and h.

Note that if g and h commute then $[g, h] = e$. To see this observe,

$$[g, h] = ghg^{-1}h^{-1} = (gh)(g^{-1}h^{-1}) = (hg)(g^{-1}h^{-1}) = h(gg^{-1})h^{-1} = heh^{-1} = hh^{-1} = e.$$

Conversely, if $[g, h] = e$ then g and h commute. See Exercise 2. Commutators are useful in mathematics wherever non-commutative operations occur.

The commutator $[g, h]$ provides a measure of how much g and h fail to commute with each other. In particular. if g and h are permutations and they fail to commute with each other by "just a little bit" then $[g, h]$ will be close to the identity, i.e. it will only permute a few numbers. This is why commutators will be of interest to us in solving permutation puzzles, they will help us to “create” good moves. You may have just realized that you frequently use “commutator moves” when solving puzzles; if this is the case then you already have a working understanding of commutators.

Example 13.1: Consider the symmetric group S_3 and the elements $s_1 = (1,2)$, $s_2 = (1,3,2)$. Then the commutator $[s_1, s_2]$ is

$$[s_1, s_2] = s_1 s_2 s_1^{-1} s_2^{-1} = (1,2)(1,3,2)(1,2)(1,2,3) = (1,3,2),$$

and the commutator $[s_2, s_1]$ is

$$[s_2, s_1] = s_2 s_1 s_2^{-1} s_1^{-1} = (1,3,2)(1,2)(1,2,3)(1,2) = (1,2,3).$$

It is not a coincidence that $[s_2, s_1] = [s_1, s_2]^{-1}$, see Exercise 3.

13.2 Creating Puzzle moves with Commutators

We will explore some properties of commutators of permutations and then see how we can apply what we learn to our standard collection of puzzles.

For a permutation $\alpha \in S_n$ define the **fixed set of** α to be the set of all numbers in $[n] = \{1, 2, 3, \ldots, n\}$ that α doesn’t move:

$$\text{fix}(\alpha) = \{m \in [n] \mid \alpha(m) = m\}.$$

The set of numbers that are not fixed by α, the ones that are moved, is the compliment of this set, which we denote by mov(α):

$$\text{mov}(\alpha) = \overline{\text{fix}(\alpha)} = \{m \in [n] \mid \alpha(m) \neq m\}.$$

By way of contrast we will refer to this as the **moved set of** α. fix(α) is precisely the set of numbers that would appear as 1-cycles in the disjoint cycle form of α, and mov(α) are those numbers that appear in cycles of length ≥ 2.[1] Since α and α^{-1} fix precisely the same objects it follows that fix(α) = fix(α^{-1}) and mov(α) = mov(α^{-1}).

In terms of permutation puzzles, mov(α) is the list of all the positions of the pieces that are moved when α is applied, and fix(α) are positions in which the pieces are left alone.

We’ll need one more bit of notation to simplify things to come. For a subset $A \subset [n]$ and a permutation $\alpha \in S_n$, we denote the set of all images of the elements of A under α as αA: [2]

$$\alpha A = \{\alpha(m) \mid m \in A\}.$$

Since α is injective then $|\alpha A| = |A|$.

Example 13.2: For $\alpha = (1,7,3,4,12)(5,9) \in S_{13}$, the set of objects that are moved is mov$(\alpha) = \{1,3,4,5,7,9,12\}$ and the set of objects that are fixed is *fix*$(\alpha) = \{2,6,8,10,11,13\}$. For $A = \{2,4,6,8,10,12\}$ and $B = \{3,7,11\}$, $\alpha A = \{\alpha(2), \alpha(4), \alpha(6), \alpha(8), \alpha(10), \alpha(12)\} = \{2,12,6,8,10,1\}$, and $\alpha B = \{\alpha(3), \alpha(7), \alpha(11)\} = \{4,3,11\}$. This can be done in Sage by using the `map` function: `map(f,L)` applies function `f` to each element of a list/set `L`.

Sage

```
sage: S13=SymmetricGroup(13)
sage: a=S13("(1,7,3,4,12)(5,9)")
sage: map(a,Set([2,4,6,8,10,12]))
[2,12,6,8,10,1]
sage: map(a,Set([3,7,11]))
[4,3,11]
```

[1] The moved set, as we have called it here, is sometimes referred to as the **support** of the permutation, denoted by supp(α).

[2] This type of set is sometimes denoted by $\alpha(A)$.

Now we are ready to investigate why the commutator $[\alpha, \beta]$ is likely to be "close" to the identity.

Let $\alpha, \beta \in S_n$, and m a number in $[n]$. If m is moved by the commutator $[\alpha, \beta]$, i.e. $m \in \text{mov}([\alpha, \beta])$, then both:

(a) $m \in \text{mov}(\alpha)$ or $\beta(m) \in \text{mov}(\alpha)$, and

(b) $m \in \text{mov}(\beta)$ or $\alpha(m) \in \text{mov}(\beta)$.

In set notation, we can write this as:

$$\text{mov}([\alpha, \beta]) \subset \left(\text{mov}(\beta) \cup \alpha^{-1}\text{mov}(\beta)\right) \cap \left(\text{mov}(\alpha) \cup \beta^{-1}\text{mov}(\alpha)\right). \tag{13.1}$$

To see why (b) is true assume that $m, \alpha(m) \notin \text{mov}(\beta)$, then $[\alpha, \beta]$ must leave m fixed:

$$[\alpha, \beta](m) = (\alpha\beta\alpha^{-1}\beta^{-1})(m) = \beta^{-1}(\alpha^{-1}(\beta(\alpha(m))) = \beta^{-1}(\alpha^{-1}(\alpha(m))) = \beta^{-1}(m) = m,$$

so $m \notin \text{mov}([\alpha, \beta])$. This proves (b). The proof of (a) is analogous.

We can describe the set of pieces that are moved in a more verbal way. First we need an alternate expression for (13.1). An equivalent way to write the set on the right-hand side in (13.1) is

$$(\text{mov}(\alpha) \cap \text{mov}(\beta)) \cup \alpha^{-1}\left(\text{mov}(\alpha) \cap \text{mov}(\beta)\right) \cup \beta^{-1}\left(\text{mov}(\alpha) \cap \text{mov}(\beta)\right).$$

This follows from the facts that $\gamma\left(\text{mov}(\delta) \cap \text{mov}(\sigma)\right) = \gamma\text{mov}(\delta) \cap \gamma\text{mov}(\sigma)$ and $\gamma^{-1}\text{mov}(\gamma) = \text{mov}(\gamma)$ (See Exercises 7 and 8). To simplify notation we will define $\text{mov}(\alpha, \beta)$ to be the intersection of $\text{mov}(\alpha)$ and $\text{mov}(\beta)$:

$$\text{mov}(\alpha, \beta) = \text{mov}(\alpha) \cap \text{mov}(\beta).$$

Therefore (13.1) can be written as

$$\text{mov}([\alpha, \beta]) \subset \text{mov}(\alpha, \beta) \cup \alpha^{-1}\text{mov}(\alpha, \beta) \cup \beta^{-1}\text{mov}(\alpha, \beta). \tag{13.2}$$

Notice $\text{mov}(\alpha, \beta)$ is the set of pieces affected by both α and β, and $\alpha^{-1}\text{mov}(\alpha, \beta)$ is the set of pieces that are moved to $\text{mov}(\alpha, \beta)$ by α, and $\beta^{-1}\text{mov}(\alpha, \beta)$ is the set of pieces moved to $\text{mov}(\alpha, \beta)$ by β. In words (13.2) says the following:

Remark 13.1. If α and β are puzzle moves, the permutation produced by $[\alpha, \beta]$ only affects pieces that are in, or moved to, locations that are moved by *both* α and β.

This remark will guide our choices for α and β. We want very little overlap in these two moves, and we want very few new pieces moved into this overlap. It can be challenging to find two moves with this property, but we can state some weaker conditions as to when $[\alpha, \beta]$ may still be a good move.

Since $|\alpha^{-1}\text{mov}(\beta)| = |\text{mov}(\beta)|$ and $|\beta^{-1}\text{mov}(\alpha)| = |\text{mov}(\alpha)|$ then (13.1) tells us that $|\text{mov}([\alpha, \beta])|$ is at most twice the size of the smaller of the sets $\text{mov}(\alpha)$ and $\text{mov}(\beta)$:

$$|\text{mov}([\alpha, \beta])| \leq 2\min\{|\text{mov}(\alpha)|, |\text{mov}(\beta)|\}. \tag{13.3}$$

So if one of $|\text{mov}(\alpha)|$ and $|\text{mov}(\beta)|$ is small, then so is $|\text{mov}([\alpha, \beta])|$. Which means $[\alpha, \beta]$ may be a good puzzle move. We can actually say something more here.

Remark 13.2. If the commutator $[\alpha, \beta]$ is to move the fewest possible pieces then α should bring as few new pieces into the locations where they will be moved by β. In other words, $\alpha^{-1}\text{mov}(\beta) \cap \text{mov}(\alpha)$ should be small.

This remark is weaker than Remark 13.1 but its conditions are sometimes easier to check in practice. With that little bit of theory behind us, let's put it into practice on a number of our favourite puzzles.

13.2.1 Rubik's Cube

Here we consider the Rubik's cube group RC_3 generated by permutations R, L, U, D, F, B. It is best if you have your Rubik's cube handy as your read through this section.

Consider the move sequence $URU^{-1}R^{-1}$. Although it is not the identity (apply it to your cube to see this), it is a lot less complex than UR alone.

Sage

```
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
sage: RC3=S48.subgroup([R,L,U,D,F,B])      # define Rubik's cube group to be RC3
```

Sage

```
sage: commutator = lambda x,y: x*y*x^(-1)*y^(-1)    # define a function called commutator
sage: commutator(U,R)
(1,3,9,33,35,27)(2,5,21)(8,24,19,43,25,30)(26,28,34)
sage: commutator(U,R).order()
6
sage: U*R
(1,38,43,19,11,35,32,30,25,17,9,48,24,8,6)(2,36,45,21,5,7,4)(3,33,27)(10\
,34,29,31,28,26,18)
sage: (U*R).order()
105
```

In the above code we defined a function called `commutator` which takes two arguments x and y and returns the product $xyx^{-1}y^{-1}$. We use a Python `lambda` function to do this, which is just a quick way to define a function in one line where no complicated decision making has to be done (see Section A.5.2 in Appendix A for more information). Of course, we really didn't need to define the function, we could have just typed in `U*R*U^-1*R^-1`, but with this function now defined we can quickly work out other commutators with less typing (just cut-and-paste).

Why should we have expected $URU^{-1}R^{-1}$ to be less complicated than UR? Many of the pieces that are moved by UR are returned to where they started by $U^{-1}R^{-1}$. For instance, consider the cubie in the ufl cubicle. The move U sends it to the ubl cubicle which is untouched by the move R, then it is moved back to the ufl cubicle by move U^{-1}, and finally move R^{-1} leave it where it is. This means the move sequence $URU^{-1}R^{-1}$ leaves the ufl cubicle untouched.

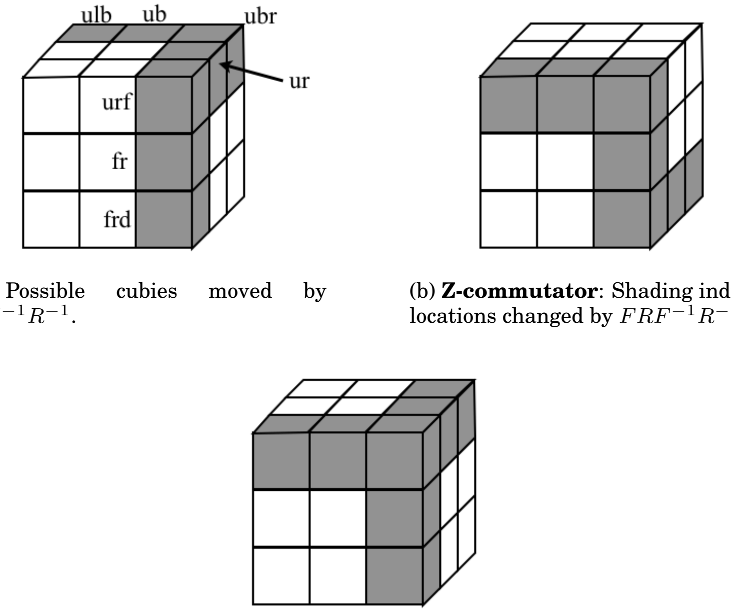
In general, if a piece is moved by U to a place that is not moved by R, then it will be moved back by U^{-1} to where it started. If the place where it started is not moved by R^{-1} – or equivalently, is not moved by R – then $URU^{-1}R^{-1}$ ends up leaving the piece where it started. Only where there is an overlap of the moves U and R are the pieces affected. The permutation produced by $URU^{-1}R^{-1}$ only affects pieces that are in, or moved to, locations common to both the *up* and *right* faces. This is precisely what (13.2) (and thus Remark 13.1) says. See Figure 13.1a.

In terms of the notation introduced, since the 3 pieces moved by both U and R are $\text{mov}(U,R) = \{urf, ur, ubr\}$, and the pieces moved to these positions by U and R are:

$$U^{-1}\text{mov}(U,R) = \{ubr, ub, ulb\} \quad \text{and} \quad R^{-1}\text{mov}(U,R) = \{frd, fr, fur\},$$

then $URU^{-1}R^{-1}$ moves at most the 7 pieces shaded in Figure 13.1a.

Commutators of two faces which share an edge occur so frequently that they have been given special names: the **Z-commutator** is $[F,R] = FRF^{-1}R^{-1}$, and the **Y-commutator** is $[F,R^{-1}] = FR^{-1}F^{-1}R$. The



(a) Possible cubies moved by $URU^{-1}R^{-1}$.

(b) **Z-commutator**: Shading indicates locations changed by $FRF^{-1}R^{-1}$

(c) **Y-commutator**: Shading indicates locations changed by $FR^{-1}F^{-1}R$

Figure 13.1: Y- and Z- commutators

names, Z-commutator and Y-commutator are used regardless of which two adjacent faces are used, all that matters is both faces are turned in the same direction (Z-commutator), or turned in opposite directions (Y-commutator). See Figure 13.1.

The cycle structure of a commutator may be such that taking powers of it will kill-off some cycles, and as a result reduce the number of pieces that are moved. This is illustrated in the next exercise.

Exercise 13.1: If x and y are basic moves of Rubik's cube associated with faces that share an edge, verify that

(a) $[x, y]^2$ permutes exactly 3 edges and does not permute any corners;

(b) $[x, y]^3$ permutes exactly 2 pairs of corners and does not permute any edges.

Let's try to create a move-sequence, using commutators, that moves only a few pieces of the cube around. Looking back at (13.2) (and Remark 13.1) we keep in mind that for any move sequences x and y, the commutator only affects pieces that are in, or moved to, locations that are moved by both x and y. For example, consider the move

$$x = LD^2L^{-1}.$$

Amongst other things, this move sequence takes ufl to bdr, and leaves all other cubies in the *up* face in their original positions. If we now consider the move

$$y = U,$$

there is only one cubie that both x and y move: the ufl cubie. Since y only moves ufr to ulf, and x only moves rbd to luf, then the only cubies that are possibly affected by $[x, y]$ are: ufl, ufr, and rbd. Trying this new move sequence out we see it moves all 3 of these cubies: the ones shaded in the Figure 13.2.The order of $[x, y] = [LD^2L^{-1}, U]$ is 3.

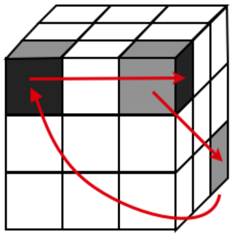


Figure 13.2: cubies moved by $[LD^2L^{-1}, U]$.

Sage

```
sage: commutator(L*D^2*L^(-1),U)
(6,8,38)(11,19,32)(17,25,48)
sage: commutator(L*D^2*L^(-1),U).order()
3
```

As another example, let's construct a move to untwist two corner cubies. Consider the two moves

$$x = L^{-1}D^2LBD^2B^{-1}, \qquad \text{and} \qquad y = U.$$

The first move may look a little complicated, but try it out for yourself. It is actually quite simple: it moves ulb to the bottom layer, then brings it back into its home location, but twisted into position bul. The only location that is affected by both x and y is ulb, but x does not move it to another location, it only twists it in place. Once x is applied, then applying y followed by x^{-1} restores the *down* and *middle* layers of the cube, and will untwist the piece that moved from ufl to ulb by y. Finally y^{-1} moves the piece that started in ufl back home, but now twisted. The result is that $[x, y]$ twists the corner piece in ufl clockwise, and the corner piece in ulb counter-clockwise as shown in Figure 13.3. When we write out the move sequence for $[x, y] = [L^{-1}D^2LBD^2B^{-1}, U]$ it is 14 moves long:

$$[x, y] = L^{-1}D^2LBD^2B^{-1}UBD^2B^{-1}LD^2L^{-1}U^{-1}.$$

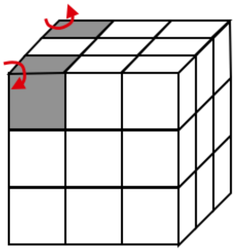


Figure 13.3: cubies moved by $[L^{-1}D^2LBD^2B^{-1}, U]$.

The move notation that we are using doesn't take into account that we can twist the whole cube around in our hands. This may make it difficult to see that a move sequence is a commutator. For example, the move sequence

$$x = U^2LR^{-1}F^2L^{-1}R$$

doesn't look like it has the form of a commutator. However, if we let $\boldsymbol{\mathcal{R}}$ denote a clockwise rotation of the whole cube around an axis through the right face, then F^2 can be written as $\boldsymbol{\mathcal{R}}U^2\boldsymbol{\mathcal{R}}^{-1}$ and so x can be seen to be the move sequence:

$$\begin{aligned} x &= U^2LR^{-1}\boldsymbol{\mathcal{R}}U^2\boldsymbol{\mathcal{R}}^{-1}RL^{-1} \\ &= [U^2, LR^{-1}\boldsymbol{\mathcal{R}}], \end{aligned}$$

which is a commutator. This move sequence is order 3 and permutes 3 edge cubies as shown in Figure 13.4. If we let M_R denote the “slice move” which consists of rotating the middle slice, parallel to the R face, in the clockwise direction, from the perspective of the R face, then we can simply write this commutator move as:

$$x = [U^2, M_R].$$

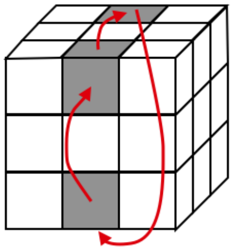


Figure 13.4: cubies moved by $[U^2, LR^{-1}\boldsymbol{\mathcal{R}}] = U^2LR^{-1}F^2L^{-1}R$.

13.2.2 Hungarian Rings

We now consider the Hungarian Rings group HR generated by permutations R and L. It is best if you have your puzzle handy (virtual or physical) as your read through this part.

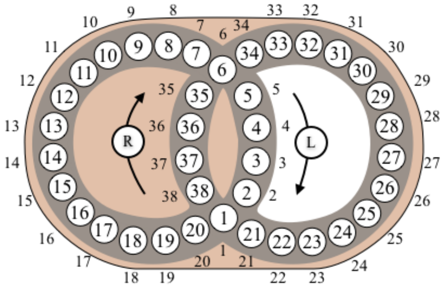


Figure 13.5: Hungarian rings puzzle.

Since each move affects over half the pieces of the puzzle then (13.3) isn’t very helpful. It says a commutator moves at most 40 pieces, but this is more than the number of pieces on the puzzle. However, using Remark 13.2 as a guide will help us create moves that affect only a few pieces.

This puzzle has the feature that the two rings intersect at only two locations (1 and 6), so the two moves L and R have very little overlap. Specifically, $\mathrm{mov}(L) = \{1, 2, 3, \ldots, 20\}$ and $\mathrm{mov}(R) = \{1, 6\} \cup \{21, 22, \ldots, 38\}$, and the intersection is $\mathrm{mov}(L, R) = \mathrm{mov}(L) \cap \mathrm{mov}(R) = \{1, 6\}$. Consequently, from (13.2) a commutator $[R^i, L^j]$, $1 \le i, j \le 19$, moves at most 6 disks:

$$\begin{aligned} \mathrm{mov}([R^i, L^j]) &= \{1, 6\} \cup R^{-i}\{1, 6\} \cup L^{-j}\{1, 6\} && (13.4)\\ &= \{1, 6, R^{-i}(1), R^{-i}(6), L^{-j}(1), L^{-j}(6)\}. && (13.5) \end{aligned}$$

Remark 13.3. On the Hungarian Rings puzzle, any commutator of the form $[L^i, R^j]$ moves at most 6 disks.

This maximum number can be reached, for example the commutator $[L, R^{-1}]$ moves 6 disks: $1, 2, 6, 7, 34, 38$.

Sage

```
sage: S38=SymmetricGroup(38)
sage: L=S38("(1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)")
```

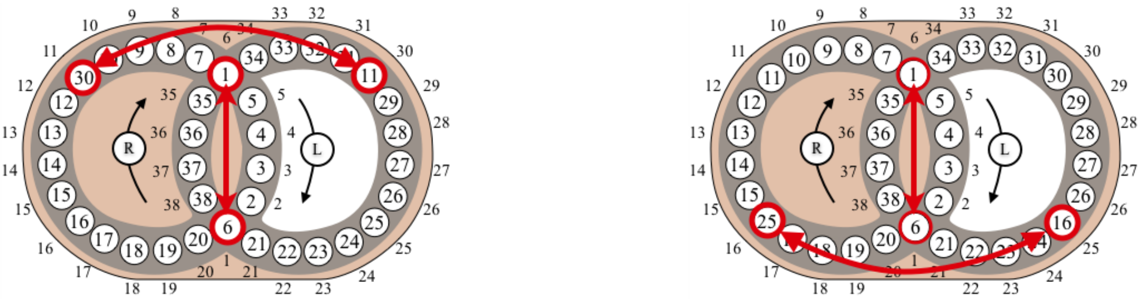
```
sage: R=S38("(1,38,37,36,35,6,34,33,32,31,30,29,28,27,26,25,24,23,22,21)")
sage: commutator(L,R^(-1))  #this is a function we defined in a previous code block
(1,38,2)(6,34,7)
```

For $[L^i, R^j]$ to move fewer than 6 disks we would need some elements in (13.5) to be the same. Remark 13.2 tells us we should look for a move L^j which moves as few new disks into spots 1 and 6 as possible. The values of j that do this are 5 and 15 (or equivalently -5). If we take $i, j \in \{5, 15\}$ then one of $L^{-i}(1) = 6$ or $L^{-i}(6) = 1$ is true, and one of $R^{-j}(1) = 6$ or $R^{-j}(6) = 1$ is true, which means mov($[R^i, L^j]$) has 4 elements. This gives the following.

Remark 13.4. On the Hungarian Rings puzzle, any commutator of the form $[L^i, R^j]$ where $i, j \in \{5, 15\}$ moves exactly 4 disks.

As an example,

$$[L^5, R^{-5}] = (1,6)(11,30), \quad \text{and} \quad [L^{-5}, R^5] = (1,6)(16,25).$$



(a) $L^5R^{-5}L^{-5}R^5$ (b) $L^{-5}R^5L^5R^{-5}$

Figure 13.6: Basic commutators on the Hungarian Rings puzzle

Knowing these commutators is enough to solve the colour version of this puzzle (see Section 16.4).

We could use Sage to determine all the powers i and j for which $|\text{mov}([L^i, R^j])| = 4$. The first line of code below defines a function `mov` whose input is a permutation `a` and whose output is the set of all numbers between 1 and n which `a` moves. The command `len(a.tuple())` just gets the value of n from the permutation in cycle form by first converting the permutation to a list, then computing its length.

Sage

```
sage: mov= lambda a: Set([ m for m in (1..len(a.tuple())) if a(m)!=m])
sage: for i in range(1,20):
sage:     for j in range(1,20):
sage:         if mov(commutator(L^(i),R^(j))).cardinality()==4:
sage:             print i, j
5 5
5 15
15 5
15 15
```

13.2.3 Oval Track Puzzle

The Oval Track group OT is generated by permutations R and T. As with the other sections, it is best if you have your puzzle handy (virtual or physical) as you read through this part.

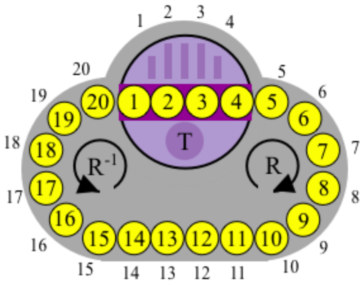


Figure 13.7: Oval Track puzzle.

A natural type of commutator to consider for this puzzle is $[R^i, T]$ where R^i is a rotation of the disks around the track by i positions, and T is a rotation of the turntable. In this case $\text{mov}(R^i) = \{1, 2, 3, \ldots, 20\}$ and $\text{mov}(T) = \{1, 2, 3, 4\}$, and so by (13.3) a commutator of this type will move at most $2\min\{20, 4\} = 8$ disks.

This maximum can sometimes be reached, for example the commutator $[R^{-4}, T] = (1, 4)(2, 3)(5, 8)(6, 7)$ moves 8 disks: $1, 2, 3, 4, 5, 6, 7, 8$.

For the commutator $[R^{-1}, T]$ the numbers of disks moved is less. This is because R^{-1} moves only one new disk into the turntable, namely disk number 5. As a result $[R^{-1}, T] = (1, 4, 2, 5, 3)$ only moves 5 disks: $1, 2, 3, 4, 5$.

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4)(2,3)")
sage: OT=S20.subgroup([R,T])
sage: commutator(R^(-4),T)    #this is a function we defined in a previous code block
(1,4)(2,3)(5,8)(6,7)
sage: commutator(R^(-1),T)
(1,4,2,5,3)
```

We will look for a commutator of the form $[R^{-i}, T]$ with a useful cycle structure. We can run a simple loop in Sage to see this quite quickly.

Sage

```
sage: for i in (1..19):
sage:     print i, commutator(R^(-i),T)
1 (1,4,2,5,3)
2 (1,4,5)(2,3,6)
3 (1,4,7)(2,3)(5,6)
4 (1,4)(2,3)(5,8)(6,7)
5 (1,4)(2,3)(6,9)(7,8)
6 (1,4)(2,3)(7,10)(8,9)
7 (1,4)(2,3)(8,11)(9,10)
8 (1,4)(2,3)(9,12)(10,11)
9 (1,4)(2,3)(10,13)(11,12)
10 (1,4)(2,3)(11,14)(12,13)
11 (1,4)(2,3)(12,15)(13,14)
12 (1,4)(2,3)(13,16)(14,15)
13 (1,4)(2,3)(14,17)(15,16)
14 (1,4)(2,3)(15,18)(16,17)
15 (1,4)(2,3)(16,19)(17,18)
16 (1,4)(2,3)(17,20)(18,19)
17 (1,18,4)(2,3)(19,20)
18 (1,20,4)(2,19,3)
```

```
19 (1,3,20,2,4)
```

For $4 \leq i \leq 16$ it is no surprise the cycle structure is a product of four disjoint 2-cycles. The commutator $[R^{-i}, T]$ brings four *new* disks: disks $i+1, i+2, i+3, i+4$, into the turntable, permutes them, then sends them back, and finally it permutes the original four disks: $1, 2, 3, 4$. The resulting permutation is:

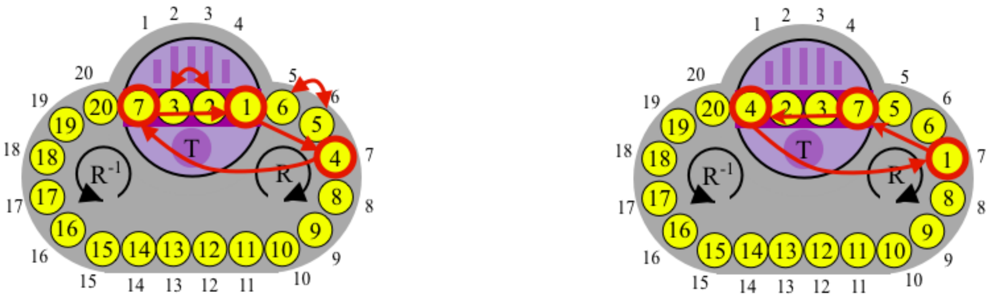
$$[R^{-i}, T] = (1,4)(2,3)(i+1, i+4)(i+2, i+3) \quad \text{for } 4 \leq i \leq 16.$$

Consider the case when $i = 1, 2, 3$. The cases when $i = 17, 18, 19$ are similar, only the rotation move R^{-i} is clockwise $20-i$ spots. Perhaps at this point we should mention why we are considering a negative exponent on R. This is really just because for $i = 1, 2, 3$, $[R^{-i}, T]$ only brings other small numbered disks into the turntable. If we were to rotate clockwise first, then some high numbered disks (i.e. $20, 19$, etc) would enter the turntable. Eventually we would like to consider variations of the puzzle where the number of disks is changed, so it would be nice to have our results expressed in such a way that do not depend on the total number of disks.

The commutator $[R^{-3}, T]$ has a particularly advantageous cycle structure, it consists of one 3-cycle and two 2-cycles. We can kill-off the 2-cycles by applying the commutator twice:

$$\begin{aligned}[R^{-3}, T]^2 &= ((1,4,7)(2,3)(5,6))^2 = (1,4,7)^2(2,3)^2(5,6)^2 \\ &= (1,7,4).\end{aligned}$$

This should be a useful move to know in solving end-game problems on this puzzle. Also, since commutators are even (see Exercise 1) this is the smallest permutation we could get using products of commutators.



(a) $R^{-3}TR^{3}T = (1,4,7)(2,3)(5,6)$

(b) $(R^{-3}TR^{3}T)^2 = (1,7,4)$

Figure 13.8: Basic commutators on the Oval Track puzzle

13.3 Exercises

1. Let $\alpha, \beta \in S_n$. Show that the commutator $[\alpha, \beta]$ is an even permutation.
2. Show that if $[g, h] = e$ then g and h commute.
3. Let G be a group and $g, h \in G$, show that $[g, h]^{-1} = [h, g]$.
4. Prove each of the following.

   (a) A permutation α commutes with the commutator $[\alpha, \beta]$ if and only if $[\alpha, \beta] = [\beta, \alpha^{-1}]$.

   (b) A permutation β commutes with the commutator $[\alpha, \beta]$ if and only if $[\alpha, \beta] = [\beta^{-1}, \alpha]$.

   (c) Both α and β commute with $[\alpha, \beta]$ if and only if $[\alpha, \beta] = [\beta, \alpha^{-1}] = [\beta^{-1}, \alpha]$.

5. We have already seen that if α and β commute then $(\alpha\beta)^n = \alpha^n\beta^n$. But this can fail if α and β do not commute. Show that if α and β satisfy the weaker hypothesis that both commute with $[\alpha,\beta]$, then for every positive integer n, $(\alpha\beta)^n = \alpha^n\beta^n[\beta,\alpha]^{n(n-1)/2}$.

6. Let $\alpha,\beta \in S_n$.

   (a) If mov(α) and mov(β) have no locations (elements) in common (i.e. $\text{mov}(\alpha)\cap\text{mov}(\beta) = \emptyset$), what is the permutation of $[\alpha,\beta]$?

   (b) If mov(α) and mov(β) have two locations (elements) in common (i.e. $|\text{mov}(\alpha)\cap\text{mov}(\beta)| = 2$), what is the largest $|\text{mov}([\alpha,\beta])|$ can be?

   (c) If mov(α) and mov(β) have two locations (elements) in common, what are the possibilities for $|\text{mov}([\alpha,\beta])|$.

7. Let $\gamma,\delta,\sigma \in S_n$. Prove the following.

   (a) $\text{mov}(\gamma) = \text{mov}(\gamma^{-1})$

   (b) $\gamma^{-1}\text{mov}(\gamma) = \text{mov}(\gamma)$

   (c) $\gamma\left(\text{mov}(\delta)\cap\text{mov}(\sigma)\right) = \gamma\text{mov}(\delta)\cap\gamma\text{mov}(\sigma)$

8. Prove that for permutations α and β,

$$\left(\text{mov}(\beta)\cup\alpha^{-1}\text{mov}(\beta)\right)\cap\left(\text{mov}(\alpha)\cup\beta^{-1}\text{mov}(\alpha)\right) = \text{mov}(\alpha,\beta)\cup\alpha^{-1}\text{mov}(\alpha,\beta)\cup\beta^{-1}\text{mov}(\alpha,\beta).$$

(Hint: Use the Distributive Law: $A\cap(B\cup C) = (A\cap B)\cup(A\cap C)$, and the results of Exercise 7.)

Rubik's Cube:

9. Find the order of the Y-commutator $[F,R^{-1}] = FR^{-1}F^{-1}R$ and of the Z-commutator $[F,R] = FRF^{-1}R^{-1}$.

10. Find the order of $[R,[F,U]]$.

11. What is the permutation produced by $[F,R^{-1}][R,U^{-1}][U,F^{-1}]$?

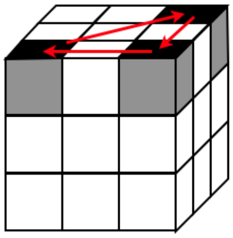
12. Show that

   (a) $[F,R^{-1}]^5 = R^{-1}[F,R]R$

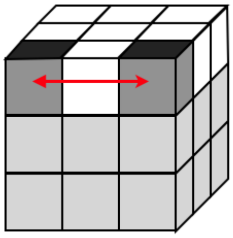
   (b) $[F^{-1},R^{-1}] = R^{-1}F^{-1}[F,R]FR$.

13. What is the permutation produced by $[(R^2U^2F^2)^3, U^2]$?

14. 3**-cycle of corners.** In this exercise you will build, as a commutator, a move which cycles 3 corner cubies as shown in the diagram.



   (a) To begin with, consider the move sequence $\alpha = F^{-1}D^{-1}FR^{-1}D^2RF^{-1}DF$. Verify that this move swaps the two corner cubes in the *up* layer, keeping their orientation (i.e. the *up* colour remains in the *up* face, which in the diagram is indicated by black). The lightly shaded cubies in the *middle* and *down* layer in the diagram move around, but the unshaded cubies remain fixed.



You may wonder how this move was constructed. The idea is to basically take one or two cubies from the *up* layer, move them to the bottom layer, perform some moves which have minimal effect on the *up* layer, then bring them back to the *up* layer. Since we don't require pieces in the *middle* and *down* layers to be returned home, coming up with these moves isn't so difficult.

(b) Since α only affects two cubies in the *up* layer, let $\beta = U$ and consider the commutator $[\alpha, \beta]$. Can you predict the effect of this move on the cubies? Hint: Remark 13.1 tells us which cubies can be affected. And with a little more thought you should be able to see how they are affected.

(c) Perform the move $[\alpha, \beta] = F^{-1}D^{-1}FR^{-1}D^2RF^{-1}DFU(F^{-1}D^{-1}FR^{-1}D^2RF^{-1}DF)^{-1}U^{-1}$ and verify your prediction from the previous part.

15. **Flip 2 adjacent edges**. Let M_R denote the "slice move" which consists of rotating the middle slice, parallel to the R face, in the clockwise direction, from the perspective of the R face. Consider the move sequence

$$\alpha = M_R U M_R^{-1} U^{-1} M_R U^2 M_R^{-1}.$$

(a) Verify α flips the edge in the fd position, and fixes everything else in the *down* layer.

(b) Since α only affects one cubies in the *down* layer, let $\beta = D$ and consider the commutator $[\alpha, \beta]$. Can you predict the effect of this move on the cubies?

(c) Perform the move $[\alpha, \beta]$ and verify your prediction from the previous part.

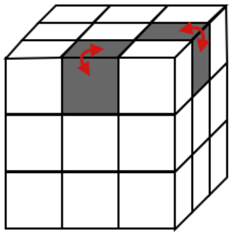
16. **Another, flip 2 adjacent edges**. If we instead would like to flip 2-edges in the *up* layer. We could consider the move sequence

$$\alpha = M_R^{-1} D M_R D^{-1} M_R^{-1} D^2 M_R.$$

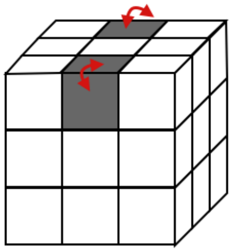
(a) Verify α flips the edge in the uf position, and fixes everything else in the *up* layer.

(b) Since α only affects one cubie in the *up* layer, let $\beta = U$ and consider the commutator $[\alpha, \beta]$. Can you predict the effect of this move on the cubies?

(c) Perform the move $[\alpha, \beta]$ and verify your prediction from the previous part. The move sequence should produce the double edge flip as shown in the figure below.



17. **Flip 2 opposite edges**. Find moves α and β so that the commutator $[\alpha, \beta]$ flips two opposite edges (as shown in the diagram below), and fixes everything else. (Hint: Modify the moves in the previous exercise.)



18. Investigate the commutators $[\alpha, \beta]$ for each of the following choices of α and β.
    (a) $\alpha = RUR^{-1}$ and $\beta = D^{-1}$
    (b) $\alpha = F^{-1}D^{-1}FR^{-1}D^2RF^{-1}DF$ and $\beta = U^2$
    (c) $\alpha = RUR^{-1}U^{-1}RUR^{-1}$ and $\beta = D^{-1}$
    (d) $\alpha = M_R^{-1}$ and $\beta = M_U^{-1}$. (M_X denotes a slice move of the middle slice parallel to face X.)

19. Create some of your own moves using commutators. Start by creating a move α which affects very few cubies in the *up* layer. Then take the commutator with $\beta = U$. Try to predict what your move with do before you even apply it.

Hungarian Rings:

20. Exploring the following commutators on the Hungarian Rings puzzle. Express the resulting permutation in disjoint cycle form.

(a) $[L, R]$	(d) $[R, L^{-1}]$	(g) $[L^5, R^{-1}]$	(j) $[R^5, L^5]$
(b) $[L, R^{-1}]$	(e) $[L^5, R]$	(h) $[L, R^{-5}]$	(k) $[L^{-5}, R^{-5}]$
(c) $[R, L]$	(f) $[R, L^5]$	(i) $[L^5, R^5]$	(l) $[L^5, R^{-5}]$

21. **Getting a 3-cycle with compound commutators.** In this exercise we investigate the compound commutator: $[[L^5, R^5], R^{-1}LR]$. It may look pretty complicated at first glance, but its construction has been well controlled. Let $\alpha = [L^5, R^5]$ and $\beta = R^{-1}LR$, so the compound commutator is $[\alpha, \beta]$. The overlap of pieces moved by both α and β consists of a single disk as we'll see below. This indicates that the commutator $[\alpha, \beta]$ will likely be a good move to know.
    (a) Show that the permutation corresponding to the commutator $\alpha = [L^5, R^5]$ is $(1, 25)(6, 11)$. Conclude that $\text{mov}(\alpha) = \{1, 6, 11, 25\}$.
    (b) Show that the only pieces of the right ring that β affects are the pieces in positions 34 and 38. Note that β affects all pieces in the left ring, except for 1 and 6. Conclude that
    $$\text{mov}(\beta) = (\text{mov}(L) - \{1, 6\}) \cup \{34, 38\} = \{38\} \cup \{20, 19, 18, \ldots, 8, 7\} \cup \{34\} \cup \{5, 4, 3, 2\}.$$
    (c) If you didn't already do so in the previous part, determine the cycle form of β.
    (d) Show that $\text{mov}(\alpha, \beta) = \{11\}$.
    (e) Show that $\alpha^{-1}\text{mov}(\alpha, \beta) = \{6\}$ and $\beta^{-1}\text{mov}(\alpha, \beta) = \{12\}$.
    (f) Conclude from formula (13.2) that $[\alpha, \beta]$ moves only $6, 11$, and 12, and verify that $[\alpha, \beta] = (6, 11, 12)$.

    Note: One could just use Sage to compute $[\alpha, \beta] = [[L^5, R^5], R^{-1}LR]$, however this wouldn't help to understand how to "build" this useful commutator in the first place. The exercises above are to get you to investigate how the commutator was constructed, so you may discover how to build your own commutators in the future.

Oval Track:

22. Determine the permutation corresponding to the commutator $[R^{-1}TR, T]$ on the Oval Track puzzle.

23. Consider the variation of the Oval Track puzzle where the turntable move T corresponds to the permutation $T = (4, 3, 2, 1)$. See Figure 13.9.

    (a) Show that $[R^{-1}, T] = (1, 2, 5)$.

    (b) Show that $[T^{-1}, R^{-1}] = (1, 5, 4)$.

    (c) Show the product $[R^{-1}, T][T^{-1}, R^{-1}]$ is $(1, 2, 4)$.

    (d) Since commutators are even, so is any product of commutators. This means that 3-cycles are the best we can do. However, the turntable move T is odd, so combining this move with a commutator may allow us to produce a 2-cycle. See what the product $[R^{-1}, T][T^{-1}, R^{-1}]T$ gives you.

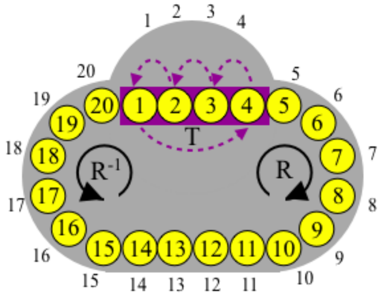


Figure 13.9: Oval Track puzzle variation for Exercise 23.

24. **Varying the turntable move T of the Oval Track puzzle.** In this exercise you will investigate, with the help of Sage , some variations of the Oval Track puzzle. In all variations[3], we assume there are 20 disks, and the usual move consisting of rotating the pieces along the track is R. We will vary the turntable move T. We have already investigated commutators on OT 1, the original Oval Track puzzle. In the previous exercise we investigated commutators on OT 2 where the turntable move is $T = (4, 3, 2, 1)$. In each case below, write out the permutation resulting from the commutator in cycle form.

    (a) $[R^{-1}, T]$ on OT 2 where $T = (4, 3, 2, 1)$

    (b) $[R^{-2}, T]$ on OT 2 where $T = (4, 3, 2, 1)$

    (c) $[T^2, R^{-2}]$ on OT 2 where $T = (4, 3, 2, 1)$

    (d) $[R^{-1}, T^2]$ on OT 2 where $T = (4, 3, 2, 1)$

    (e) $[R^{-1}, T]$ on OT 4 where $T = (5, 4, 3, 2, 1)$

    (f) $[R^{-1}, T^2]$ on OT 4 where $T = (5, 4, 3, 2, 1)$

    (g) $[R^{-1}, T]$ on OT 5 where $T = (1, 2)(3, 4)$

    (h) $[R^{-2}, T]$ on OT 5 where $T = (1, 2)(3, 4)$

    (i) $[R^{-3}, T]$ on OT 5 where $T = (1, 2)(3, 4)$

    (j) $[R^{-5}, T]$ on OT 17 where $T = (1, 6)(2, 5)(3, 4)$

[3] Variation names are due to John O. Kiltinen who studies these in his book [9].

Lecture 14

Conjugates

Commutators provided us with a method for creating puzzle moves that affect only a small number of pieces. In this lecture we introduce "conjugation" which is a process for modifying existing moves to produce new moves that have similar structure.

14.1 Conjugates

When playing with permutation puzzles, a move sequence of the form "move 1, then move 2, then inverse of move 1" comes in handy. Moves of this form are called *conjugates*. You may have just realized that you frequently use "conjugate moves" when solving puzzles, if this is the case then you already have a working understanding of conjugation. As you read through this lecture, you will find it useful to have a puzzle on hand to try things out for yourself.

Definition 14.1: If g, h are two elements of a group G, then we call the element

$$g^h = h^{-1}gh$$

the **conjugate** of g by h.

Note that $g^h = g$ if and only if g and h commute. Therefore, much like the commutator, the conjugate g^h provides a measure of how much g and h fail to commute with each other. If g and h don't commute, then $g^h \neq g$, however g^h should be like g in some ways. In the case of permutations we can say exactly how they are similar. This is stated in Lemma 14.1 below.

The exponential notation is used because conjugation enjoys similar properties to that of exponentiation. See Exercise 5.

Example 14.1: Consider the symmetric group S_3 and the elements $s_1 = (1, 2)$, $s_2 = (1, 3, 2)$. Then the conjugate of s_1 by s_2 is

$$s_1^{s_2} = s_2^{-1}s_1s_2 = (1, 2, 3)(1, 2)(1, 3, 2) = (1, 3)$$

and the conjugate of s_2 by s_1 is

$$s_2^{s_1} = s_1^{-1}s_2s_1 = (1, 2)(1, 3, 2)(1, 2) = (1, 2, 3).$$

Definition 14.2: We say that two elements $g_1, g_2 \in G$ are **conjugate** (in G) if there is an element $h \in G$ such that $g_2 = g_1^h$.
The set of all elements in G that are conjugate to g is called the **conjugacy class of g** and denoted by $\mathrm{cl}(g)$:

$$\mathrm{cl}(g) = \{xgx^{-1} \mid x \in G\}.$$

In the example above we see that cycle structure seems to be preserved by conjugation. By this we mean, the conjugate of a 2-cycle is also a 2-cycle, the conjugate of a 3-cycle is also a 3-cycle. This is true in general, and we state it as the following remark. We prove this as a part of the subsequent lemma.

Remark 14.1. For $\alpha, \beta \in S_n$, the two permutations α and $\beta^{-1}\alpha\beta$ have the same cycle structure.

Lemma 14.1 (Conjugation preserves cycle structure): *Let α, β be any permutation in S_n, and suppose $\alpha(i) = j$. Then $\alpha^\beta = \beta^{-1}\alpha\beta$ sends $\beta(i)$ to $\beta(j)$:*

$$(\alpha^\beta)(\beta(i)) = \beta(j).$$

Moreover, if α has cycle structure

$$\alpha = (a_1, a_2, \ldots, a_{k_1})(b_1, b_2, \ldots, b_{k_2}) \cdots (c_1, c_2, \ldots, c_{k_m})$$

then α^β has the same cycle structure

$$\alpha^\beta = (\beta(a_1), \beta(a_2), \ldots, \beta(a_{k_1}))(\beta(b_1), \beta(b_2), \ldots, \beta(b_{k_2})) \cdots (\beta(c_1), \beta(c_2), \ldots, \beta(c_{k_m}))$$

Proof: To see $\alpha^\beta = \beta^{-1}\alpha\beta$ sends $\beta(i)$ to $\beta(j)$ we notice

$$\alpha^\beta(\beta(i)) = (\beta^{-1}\alpha\beta)(\beta(i)) = \beta(\alpha(\beta^{-1}(\beta(i)))) = \beta(\alpha(i)) = \beta(j).$$

To show that the cycle structure is as described in the statement of the lemma first express α in disjoint cycle form: $\alpha = \sigma_1\sigma_2\cdots\sigma_m$, where σ_i is a k_i-cycle. Observe that

$$\alpha^\beta = \beta^{-1}(\sigma_1\sigma_2\cdots\sigma_m)\beta = (\beta^{-1}\sigma_1\beta)(\beta^{-1}\sigma_2\beta)\cdots(\beta^{-1}\sigma_m\beta),$$

so it suffices to prove the result for each of the cycles σ_i.

Consider the cycle $\sigma = (a_1, a_2, \ldots, a_k)$, and let $d_i = \beta(a_i)$. By the first part of the lemma, which we have already proved, σ^β contains the cycle $(d_1, d_2, \ldots, d_k)$. Moreover, if x is an element that is moved by σ^β then $(\beta^{-1}\sigma\beta)(x) \neq x$ and so $\sigma(\beta^{-1}(x)) \neq \beta^{-1}(x)$, which means $\beta^{-1}(x) = a_i$ for some i. Therefore, $x = d_i$ for some i. It follows that

$$\sigma^\beta = (d_1, d_2, \ldots, d_k).$$

This proves the lemma. □

As an example, this lemma and the preceding remark tells us that if we have a 3-cycle α, then no matter what the permutation β is, the conjugate α^β will be another 3-cycle. This is how we will use conjugation to modify existing puzzle moves.

As a consequence of Lemma 14.1 it is easy to see when two permutation $\alpha, \beta \in S_n$ are conjugate in S_n: they are conjugate if and only if the cycles in their respective disjoint cycle forms have the same length when arranged from shortest to longest (i.e. they have the same cycle structure). The "only if" part we have already proven. On the other hand, if two permutation α and β have the same cycle structure then arrange their disjoint cycle forms as follows (here we insert 1-cycles on the end):

$$\begin{aligned}\alpha &= (a_{1,1}, a_{1,2}, \ldots, a_{1,k_1})(a_{2,1}, a_{2,2}, \ldots, a_{2,k_2}) \cdots (a_{m,1}, a_{m,2}, \ldots, a_{m,k_m})(a_{m+1,1}) \cdots (a_{m+s,1})\\ \beta &= (b_{1,1}, b_{1,2}, \ldots, b_{1,k_1})(b_{2,1}, b_{2,2}, \ldots, b_{2,k_2}) \cdots (b_{m,1}, b_{m,2}, \ldots, b_{m,k_m})(b_{m+1,1}) \cdots (b_{m+s,1})\end{aligned}$$

and construct a permutation γ such that $\gamma(a_{i,j}) = b_{i,j}$. It follows the $\alpha^\gamma = \beta$ and so α and β are conjugate. (Note, γ is not necessarily unique.)

For example, the permutations

$$\alpha = (1,2,3)(4,5,6,7,8)(9,10), \quad \text{and} \quad \beta = (4,5,3)(1,8,2,10,11)(7,12)$$

are conjugate in S_{12}. One possibility for γ is $(1,4)(2,5,8,11,6)(3)(7,10,12,9)$.

14.2 Modifying Puzzle moves with Conjugates

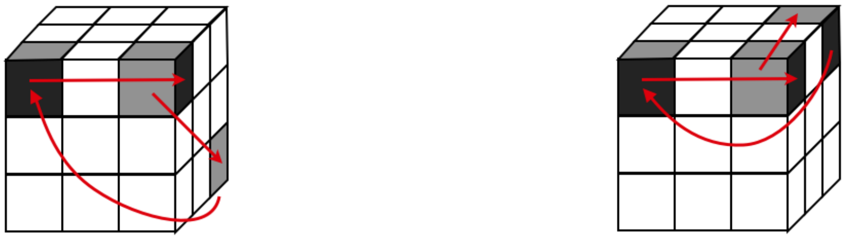
We've already made extensive use of conjugation while investigating the 15-puzzle. We showed in Lecture 9 that the solvable configurations of the 15-puzzle, where the empty space is in box 16, are precisely the even permutations. The way we argued this was we found one 3-cycle, namely $(11,12,15)$ and by conjugation we were able to modify this to produce any other 3-cycle.

In general, if we have a move α that does something useful then we can modify it by using conjugates. First find a set-up move β^{-1} that takes some pieces that we wish to affect and moves them to the positions affected by α. Applying α then affects these new pieces, and β then moves everything back. This will only affect the pieces moved by β^{-1} into mov(α), and they are permuted with the same structure of α. This description may seem a little confusing, but once you've played around with conjugates you will see their actions are very intuitive. We'll look at many examples for the various puzzles over the next few sections.

14.2.1 Rubik's Cube

It is best if you have your Rubik's cube in hand while reading through this part.

Looking back at the commutators we constructed in Lecture 12 you will notice that many of the x moves were made up of conjugates. We saw that the commutator $[LD^2L^{-1}, U]$ permuted three corner cubies as shown in Figure 14.1a.



(a) 3-cycle of corner cubies by commutator $LD^2L^{-1}ULD^2L^{-1}U^{-1}$

(b) conjugation of commutator by B

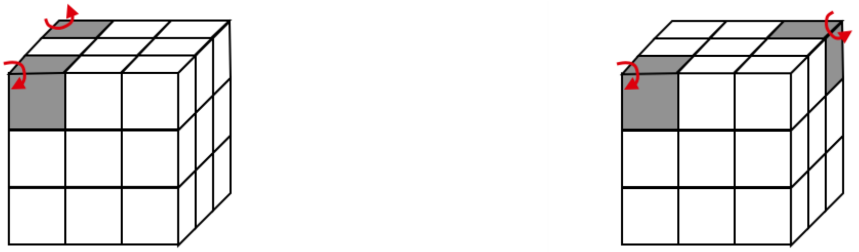
Figure 14.1: cycling 3 corner cubies

We will modify this move so it permutes the three corner cubies as show in Figure 14.1b. To do this, first apply the set-up move B^{-1} which takes the urb corner piece to the rdb position. Then applying commutator $[LD^2L^{-1}, U]$ cycles the 3 corner cubies as shown in Figure 14.1a, though the piece in the rdb position is really the piece that started in the urb position. Undoing the set-up move results in a move sequence $B^{-1}[LD^2L^{-1}, U]B$ which moves the cubes as shown in Figure 14.1b.

As another example, the commutator $[x, y]$ where

$$x = L^{-1}D^2LBD^2B^{-1}, \qquad \text{and} \qquad y = U$$

produced a twist of 2 corners as shown in Figure 14.2a. If we use the set-up move B, before apply the corner twist commutator, then undo the set-up move by taking B^{-1}, then we produce a new move which twists diagonally opposite corner cubies (see Figure 14.2b). This new move is the conjugate $B[L^{-1}D^2LBD^2B^{-1}, U]B^{-1}$.



(a) 2 corner twist by commutator $xyx^{-1}y^{-1}$

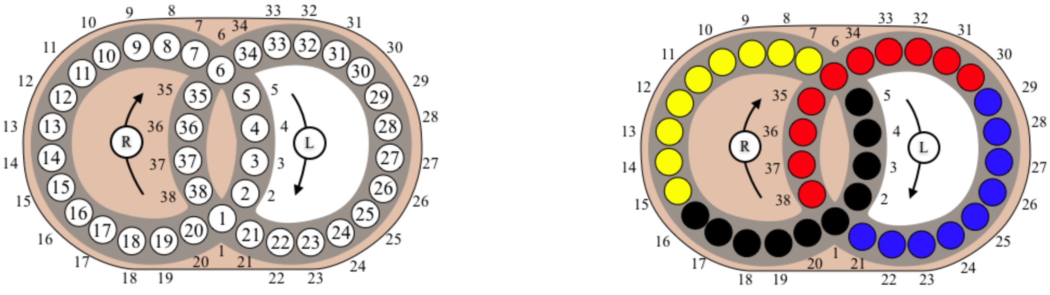
(b) conjugation of commutator by B^{-1}

Figure 14.2: twisting 2 corner cubies

14.2.2 Hungarian Rings

Using commutators we found some very useful moves on the Hungarian Rings puzzle:

$$[L^5, R^5] = (1, 25)(6, 11), \qquad [L^{-5}, R^{-5}] = (1, 16)(6, 30),$$
$$[L^5, R^{-5}] = (1, 6)(11, 30), \qquad [L^{-5}, R^5] = (1, 6)(16, 25).$$



(a) Hungarian Rings with numbers

(b) Hungarian Rings with colours

Figure 14.3: Hungarian Rings puzzle

Suppose we wanted to swap the contents of boxes 8 and 27, then we could move 8 to position 11, 27 to position 30. Call the move that does this γ. For example $\gamma = L^{-3}R^{-3}$ would achieve this. Then apply the commutator $[L^5, R^{-5}]$, which swaps disks 8 and 27, along with the disks in positions 1 and 6. Now undo the set-up move γ. The move sequence performed is $\gamma[L^5, R^{-5}]\gamma^{-1}$. Since the cycle structure is the same as $[L^5, R^{-5}]$ it swapped two pairs of disks, one pair being 8 and 27, and the other one was $32, 36$.

In the coloured version of the Hungarian Rings puzzle, shown in Figure 14.3b, if two balls of the same colour are in the intersection positions (1 and 6) then applying one of the commutators, say $[L^5, R^{-5}] =$

$(1,6)(11,30)$, would swap balls in positions 11 and 30, but the 1, 6-swap would go unnoticed since the balls were identical. This gives us a way to swap any two balls on the puzzle, and as we know from the theory of permutations, this is enough to construct any permutation of the coloured balls.

In the numbered version of the puzzle, where every ball is distinct (Figure 14.3a) this is still not enough to solve every permutation. We will actually need to find a genuine 2-cycle. We'll pick up this topic in Section 16.2.4 of Lecture 16. Though, armed with the tools of commutators and conjugates perhaps you can discover such a move for yourself! Next we'll use commutators to construct a 3-cycle.

Compound Commutator - Getting a 3-cycle.

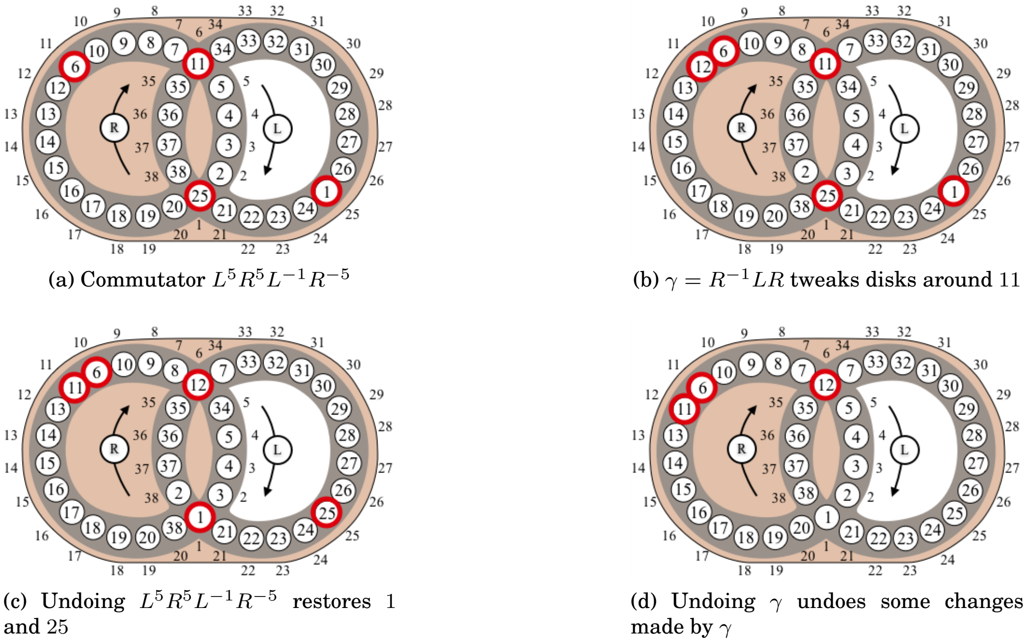
We have seen that by using commutators we can produce a product of two disjoint 2-cycles. For example $[L^5, R^5] = (1,25)(6,11)$. We now show that we are able to produce a move sequence which gives a 3-cycle by using *compound commutators*, that is, something of the form:

$$[[\alpha,\beta],\gamma] = (\alpha\beta\alpha^{-1}\beta^{-1})\gamma(\beta\alpha\beta^{-1}\alpha^{-1})\gamma^{-1}.$$

Since one of the transpositions in $[L^5, R^5]$ involves the lower point of intersection (position 1) and the right ring, while the other involves the upper point of intersection (position 6) and the left ring, we should be able to tweak one of the intersection points while leaving the other unchanged. We would like a move γ that has little overlap with $[L^5, R^5]$, where mov($[L^5, R^5]$) $= \{1, 6, 11, 25\}$. Since each ring has 3 disks which are moved by $[L^5, R^5]$ we would like a move that temporarily moves the disks out of the intersection points, then moves the left ring (for example), and then moves disks back onto the intersection points. It would then follow that mov($[L^5, R^5]$) $\cap$ mov(γ) $= \{11\}$. Consider the move $\gamma = R^{-1}LR$. This leaves the disks in positions 1, 6 and 25 unchanged, but it moves the disk in position 11 to position 10. See Figures 14.4a and 14.4b. The circled positions in the figure are just to draw you attention to these positions. The pieces affected by the commutator $[[L^5, R^5], \gamma]$ are at most

$$\begin{aligned}\text{mov}([[L^5,R^5],\gamma]) &\subset \text{mov}([L^5,R^5],\gamma) \cup [L^5,R^5]^{-1}\text{mov}([L^5,R^5],\gamma) \cup \gamma^{-1}\text{mov}([L^5,R^5],\gamma)\\ &= \{11\} \cup [L^5,R^5]^{-1}\{11\} \cup \gamma^{-1}\{11\}\\ &= \{11, 6, 12\}\end{aligned}$$

In fact, $[[L^5, R^5], \gamma]$ is the 3-cycle $(6, 11, 12)$.



(a) Commutator $L^5R^5L^{-1}R^{-5}$

(b) $\gamma = R^{-1}LR$ tweaks disks around 11

(c) Undoing $L^5R^5L^{-1}R^{-5}$ restores 1 and 25

(d) Undoing γ undoes some changes made by γ

Figure 14.4: A compound commutator that uses the commutator $[L^5, R^5]$ to construct a 3-cycle $(6, 11, 12)$

14.2.3 Oval Track Puzzle

Conjugation is a very natural process on the Oval Track puzzle. If you have spent some time playing with the puzzle you undoubtedly use conjugation on almost every move. The reason for this is the turntable is located on one part of the puzzle, pieces will need to be moved into the turntable say by a move R^j, then they are rotated in the turntable T, and finally the pieces are moved back R^{-j}. The move sequence R^jTR^{-j} is conjugation.

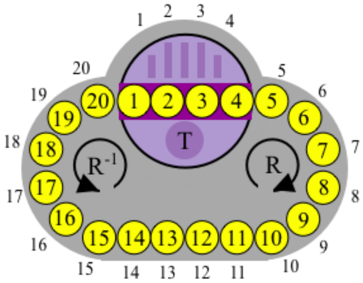
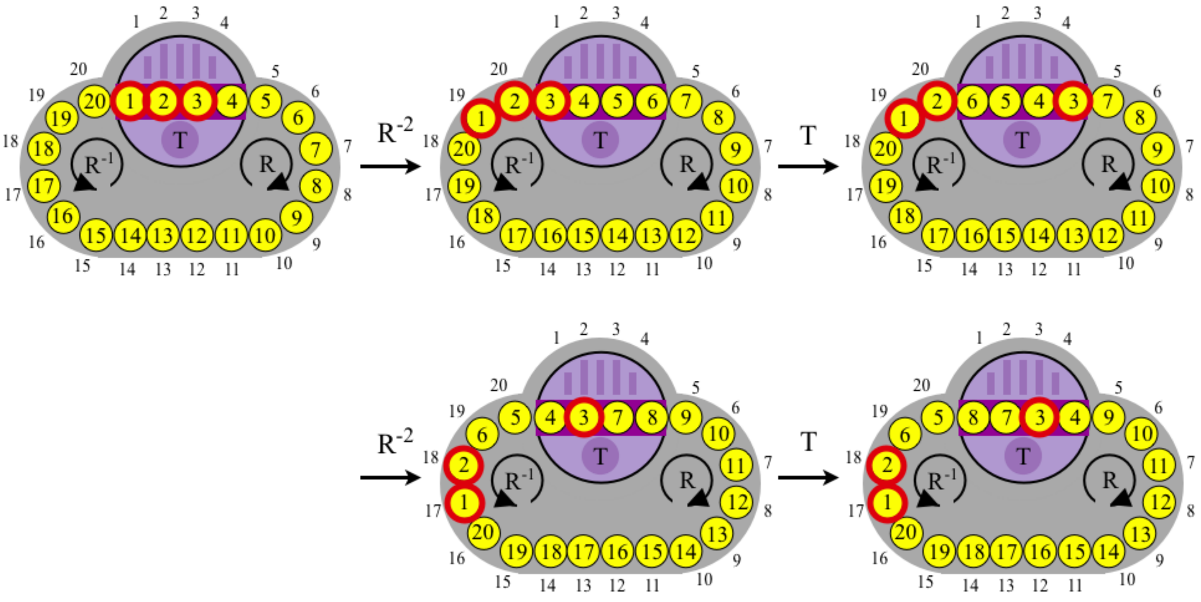


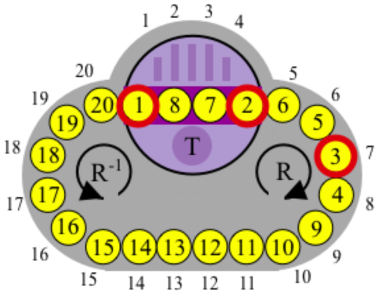
Figure 14.5: Oval Track puzzle.

Using commutators we found the 3-cycle $\gamma = [R^{-3}, T]^2 = (1, 7, 4)$. Any conjugate of this would also be a 3-cycle, so let's try to construct the 3-cycle $(1, 2, 3)$. To do this we would need to find a move sequence β that takes $\{1, 2, 3\}$ to $\{1, 7, 4\}$. The order doesn't matter much, for example we could find a move sequence that takes $1 \mapsto 1$, $2 \mapsto 4$, and $3 \mapsto 7$. What is important though is once we get $1, 2, 3$ into positions $1, 4, 7$ then we must cycle them appropriately: either γ or γ^{-1}. So before we do anything we make a mental note that to produce the 3-cycle $(1, 2, 3)$ we want "1 to chase 2". By this we mean, once we get disks $1, 2, 3$ into positions $1, 4, 7$ we then cycle them in the direction so the 1 goes to the current position of 2. The rest of the disks will follow accordingly.

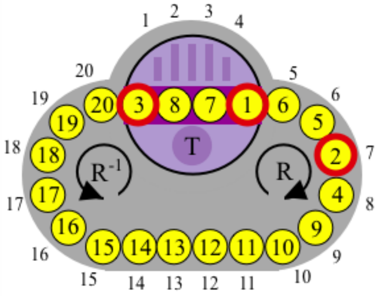
Since 1 is already in position 1 we leave it there. The move β just needs to take $2 \mapsto 4$ and $3 \mapsto 7$. We begin by pushing disk 3 away from the rest of the pack. To do this, move it to position 1 and apply T. It stills need to move one more unit to the right in order to be 6 units away from disk 1, so move it to position 2 and apply T. This move sequence $R^{-2}TR^{-2}T$ has now pushed disk 3 far enough away from 1 so that if 1 rotates to its home position disk 3 will be in position 7. See the following figure.



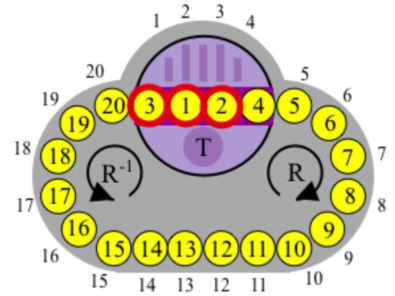
Now, using the space between disks 1 and 3 we push 2 two units to the right. This is done by putting it in position 1 applying T, then putting it again in position 3 and applying T. The move sequence to do this is $R^3TR^{-1}T$. The complete move sequence is $\beta = R^{-2}TR^{-2}TR^3TR^{-1}TR^2$, and the puzzle now looks like this:



We now apply the 3-cycle $\gamma = [R^{-3}, T]^2 = (1, 7, 4)$, but we have to recall we wanted 1 to chase 2. Since 2 is in position 4 we want to apply the 3-cycle $(1, 4, 7)$ which is actually γ^{-1}. After applying γ^{-1} the puzzle now looks like this:



Finally, undoing β returns all pieces back to their original positions, except the pieces circled in red. These pieces have been moved since β was applied. β^{-1} will take the piece in position 1 back to 1, the piece in position 4 back to 2, and the piece in position 7 back to 3.



Therefore the move sequence $\beta\gamma\beta^{-1}$ produces the 3-cycle $(1, 2, 3)$.

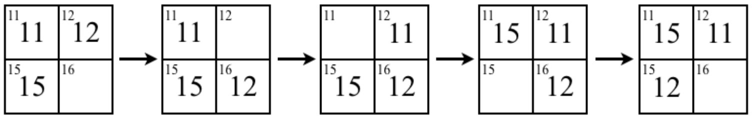
14.2.4 15-Puzzle

Here we revisit our results about the 15-puzzle using our new tool: conjugation. The proof of the solvability criteria, Theorem 9.1, which states that

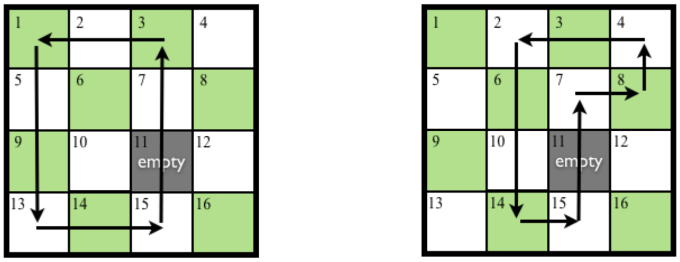
> A configuration of the tiles, in which the empty space is in box 16, is solvable if and only if it is an even permutation.

relied on the ability to construct 3-cycles. The essence of the proof was based on conjugation.

Recall we can produce the 3-cycle $\sigma = (11, 12, 15)$ by focussing on the bottom right corner of the puzzle:



From this one 3-cycle σ, we can conjugate it to construct any other 3-cycle we want. To do this we just need a way to move any 3 tiles down to the bottom right-hand corner, along with the empty space. Hiding any tiles you have already brought down in boxes 12 and 15, we can bring any other tile down using one of the two tours in Figure 14.6. Call the move sequence which brings all three tiles down β.



(a) (b)

Figure 14.6: Tours for producing 3-cycles.

Applying σ cycles the tiles around (or if you want to cycle them in the other direction apply σ^{-1}). Then β^{-1} takes all the tiles back to where they started, but with the main three tiles now cycled. In other words $\beta\sigma\beta^{-1}$ is precisely the move that cycles the three tiles.

This was a purely theoretical argument. In practice solving the puzzle in this way is completely inefficient. However, if one wants to produce a particular 3-cycle it is not necessary to push the 3 tiles down to the bottom right-hand corner, apply $(11, 12, 15)$, then reverse the moves. Instead, if we can apply a sequence of moves β which take the 3 tiles, along with the empty space, into any 2-by-2 array then we can perform a 3-cycle there, call it σ, then apply β^{-1}. The resulting move sequence $\beta\sigma\beta^{-1}$ will be a 3-cycle on the selected tiles.

14.3 Exercises

1. For each of the pairs of permutations $\alpha, \beta \in S_n$ calculate the conjugate $\alpha^{-1}\beta\alpha$. Note that it has the same cycle structure as β, and notice the each entry in the cycle is the image under α of the corresponding entry in the cycle of β.
   (a) $\alpha = (1, 2, 3, 4, 5, 6, 7)(8, 9, 10), \qquad \beta = (1, 5, 8)(2, 6)(3, 7, 4)$
   (b) $\alpha = (1, 5, 8)(2, 6)(3, 7, 4), \qquad \beta = (1, 2, 3, 4, 5, 6, 7)(8, 9, 10)$
   (c) $\alpha = (1, 7, 5, 9, 3, 10, 12)(4, 6)(8, 11), \qquad \beta = (1, 6)(2, 8)(4, 7)$
2. For $\alpha = (1, 2, 3, 4)(5, 6)$ and $\beta = (1, 6)(2, 5, 3)$ do the following.
   (a) Calculate $\beta^{-1}\alpha\beta$.
   (b) Calculate the values of $\beta(1)$, $\beta(2)$, $\beta(3)$, $\beta(4)$, $\beta(5)$, $\beta(6)$, then write down the product of cycles, $(\beta(1), \beta(2), \beta(3), \beta(4))(\beta(5), \beta(6))$.
   (c) Observe that the product of cycles in part (b) is the same as the answer to part (a). This is the essence of Lemma 14.1.
3. For each of the following pairs of permutations state whether they are conjugate in S_{10}. That is, determine whether there exists a $\gamma \in S_{10}$ so that $\alpha = \gamma^{-1}\beta\gamma$.
   (a) $\alpha = (1, 2, 3, 4, 5, 6, 7)(8, 9, 10), \qquad \beta = (1, 5, 8)(2, 6, 3, 7, 4, 10, 9)$
   (b) $\alpha = (1, 5, 8)(2, 6)(3, 7, 4), \qquad \beta = (1, 2)(7, 3)(8, 9, 10)$
   (c) $\alpha = (1, 7, 5, 9, 3), \qquad \beta = (1, 6, 2, 8, 4)$
4. Let G be a group. Prove that every conjugate of a commutator is a commutator by showing that $g[a, b]g^{-1} = [gag^{-1}, gbg^{-1}]$ for all $a, b, g \in G$.
5. Show that for $g_1, g_2, h, h_1, h_2 \in G$ the following hold.
   (a) $(g_1 g_2)^h == g_1^h g_2^h$
   (b) $g^{h_1 h_2} = (g^{h_1})^{h_2}$
6. Show that g and g^h have the same order.
7. For permutations $\alpha, \beta \in S_n$, show that α and α^β have the same parity.
8. Show that the notion of conjugate defines an equivalence relation. That is, show that
   (a) any element of $g \in G$ is conjugate to itself (**reflexive**)
   (b) if g is conjugate to h, then h is conjugate to g (**symmetry**)
   (c) if g is conjugate to h, and h is conjugate to k, then g is conjugate to k (**transitivity**)
9. Show that the conjugacy classes form a partition of G. That is, show that G can be expressed as a disjoint union of distinct conjugacy classes.

10. **Is the building of commutators associative?** (a) Explore the equation $[[\alpha, \beta], \gamma] = [\alpha, [\beta, \gamma]]$ by trying out these compound commutators on one of the puzzles. Show that this equation is not true for all permutations α, β, and γ. This show the operation of commutator building is not an associative operation. (b) Show that for any permutations α, β, and γ such that β commutes with both α and γ, this associativity equation is trivially true.

11. The expressions $[(\alpha\beta), \gamma]$ and $[\alpha, (\beta\gamma)]$ are commutators of products. Prove the following formulas which show a commutator of products is a product of commutators.

    (a) $[\alpha, (\beta\gamma)] = [\alpha, \beta](\beta[\alpha, \gamma]\beta^{-1}) = [\alpha, \beta][\beta\alpha\beta^{-1}, \beta\gamma\beta^{-1}]$

    (b) $[\alpha\beta, \gamma] = (\alpha[\beta, \gamma]\alpha^{-1})[\alpha, \gamma] = [\alpha\beta\alpha^{-1}, \alpha\gamma\alpha^{-1}][\alpha, \gamma]$

15-Puzzle:

12. Starting with the 15-puzzle in the solved state write down a sequence of moves which will produce each of the following 3-cycles.

    (a) $(2, 12, 7)$ (b) $(3, 8, 12)$ (c) $(3, 9, 13)$.

    Either write the moves using transpositions, or use the words “up”, “down”, “left”, “right”, to indicate the direction the tile adjacent to the empty space is moved. Rather than bringing the three tiles together in the lower right-hand corner, bring them together with the empty space into any 2-by-2 array that is convenient. (It may help to use a physical or virtual version of the puzzle, see the “software” section of [10] for some links.)

Rubik’s Cube:

13. **Set-up Moves.** The move β^{-1} in the conjugate $\beta^{-1}\alpha\beta$ is called a *set-up* move. This is because it is the move that brings the desired pieces into the positions that are affected by α, once α is applied, the pieces are then restored by applying β. The important thing to keep in mind with these set-up moves is that it doesn’t matter how the other pieces are moved around, this will eventually be undone. All that matters is how a small subset of pieces are moved, this is where we are to focus our attention. To get some practice in creating set-up moves, find a sequence of moves which accomplishes each of the following. In each case your move sequence should move the pieces as described, and we don’t really care if it moves the other cubies that aren’t mentioned. (See comment below for an explanation of the notation used.)

    (a) Moves the piece in the urf corner to the frd position.

    (b) Moves the piece in the rdf corner to the fur corner.

    (c) Moves the piece in the ur edge to the ul edge position, and the piece in the ul edge to the ur edge position.

    (d) Moves the piece in the ur edge to the ul edge position, and the piece in the ul edge to the ru edge position.

    (e) Moves the piece in the uf edge to the fu edge position (i.e. it flips the uf edge piece).

    (f) Moves the piece in the ufr corner to the fru corner position (i.e. it rotates the ufr corner piece counterclockwise).

    (g) Moves the piece in the ufr corner to the ulb corner, and the piece in the ulb corner to the ufr corner.

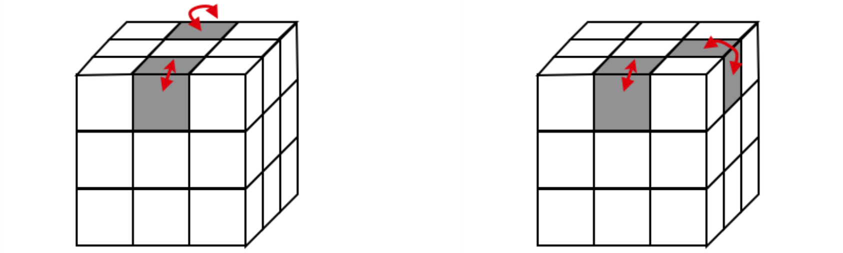
    Notation: Here order of how the positions are listed matters. For example $urf \mapsto rdf$ means the corner which is part of the *up*, *right*, and *front* faces is moved to the corner which is part of the *right*, *down*, and *front* faces, and moreover the facet in the *up* face moves to the *right* face, the facet in the *right* face moves to the *down* face, and the facet in the *front* face moves to the *front* face.

14. **More Set-up moves.** Find a sequence of moves which accomplishes the following:

$$ufr \mapsto bur, \quad bur \mapsto lfu, \quad lfu \mapsto ufr.$$

Do this so that all other cubies in the *up* layer remain in their home positions, but all other cubies in the *middle* and *down* layer may move around.

15. Suppose we know a move α which flips two opposite edges in the top layer, as shown in Figure 14.7a. Find a move β so the $\beta^{-1}\alpha\beta$ flips two adjacent edges in the top layer, as shown in Figure 14.7b.

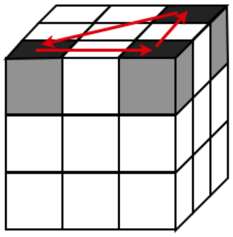


(a) Known move α flips opposite edges. (b) Determine how to achieve this move from α.

Figure 14.7: Exercise 15

16. The commutator $[L^{-1}D^2L, U]$ permutes corner cubies as follows (ulb, ufl, frd). (Here we are using our cycle notation as a compact way to represent the movement of pieces.) What pieces does the conjugate $R^{-1}[L^{-1}D^2L, U]R$ permute?

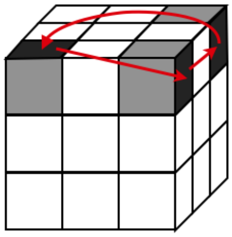
17. **Building a corner 3-cycle.** In this exercise we build a 3-cycle of corners in the *up* layer that preserves orientation (that is, the *up* facets remain in the *up* layer for each of the corners cubies being moved). The desired movement is shown in the figure, where black facets are moved to black facets.



(a) Verify the conjugate ULU^{-1} brings one new corner cubie into the *right* face.

(b) Since ULU^{-1} brings one new corner cubie into the *right* face this makes a good candidate to form a commutator with R^{-1}. Verify the commutator $[ULU^{-1}, R^{-1}]$ moves the corner cubies as indicated in the diagram. The movement of pieces is also given notationally as follows

$$ulf \mapsto ruf, \quad ruf \mapsto rbu, \quad rbu \mapsto ulf.$$



(c) Unfortunately, the commutator $[ULU^{-1}, R^{-1}]$ twists the corners in addition to permuting them. We'd like to tweak this commutator a little bit so the it doesn't twist the corners. Find a set-up move γ which twists some of the corners in place, so that when the commutator $[ULU^{-1}, R^{-1}]$ is applied, followed by γ^{-1}, the corner cubies that were permuted still have their *up* facets in the *up* layer.

(Hint: a move which twists ufl counterclockwise, and urb clockwise, and leaves ufr alone (and possibly moving other pieces) should work.)

(d) Verify that $\gamma[ULU^{-1}, R^{-1}]\gamma^{-1}$ produces the desired 3-cycle of corners (as shown in the first figure above).

Oval Track:

18. By conjugating the 3-cycle $(1, 4, 7)$ produce three other 3-cycles, say $(1, 2, 4)$, $(2, 8, 14)$, and $(5, 10, 15)$.

19. (a) Verify that TR^{-1} is the product of a 17-cycle and a 2-cycle.

(b) By raising TR^{-1} to the power of 17 the 17-cycle can be killed-off, leaving just a 2-cycle. Verify that $(TR^{-1})^{17} = (1, 3)$.

(c) Find a move sequence β so that $\beta(TR^{-1})^{17}\beta^{-1} = (1, 2)$.

(d) Using conjugation produce two other 2-cycles on this puzzle, say $(5, 15)$ and $(9, 12)$.

(e) Convince yourself that you can produce any 2-cycle as a conjugate of $(TR^{-1})^{17}$. Since every permutation is a product of 2-cycles you have proven that every permutation is obtainable in this puzzle.

Lecture 15

Mastering the Oval Track Puzzle

We now have enough theory developed to give a thorough analysis of the Oval Track puzzle.

15.1 Oval Track with $T = (1,4)(2,3)$

In this section we focus on the standard Oval Track puzzle as shown in Figure 15.1. This version is also known as TopSpin and was once manufactured by Binary Arts (now ThinkFun).

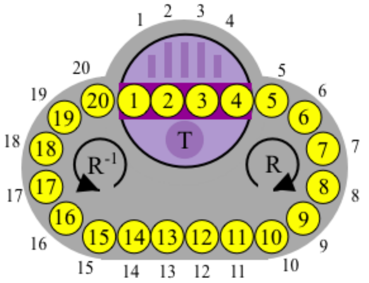


Figure 15.1: Oval Track puzzle.

The two basic moves of the Oval Track puzzle are R, and T, where R denotes a clockwise rotation of numbers around the track, where each number moves one space, and T denotes a rotation of the turntable.

The permutation corresponding to the legal moves R and T are as follows:

$$R = (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)$$
$$T = (1,4)(2,3)$$

and the Oval Track puzzle group is $OT = \langle R, T\rangle$.

Note that $T^{-1} = T$ since T has order 2, and R^{-1} represents a counterclockwise rotation of the disks along the track.

Let's get right down to business and find out which permutations of the 20 disks are possible. We can set-up the corresponding puzzle group OT in Sage and compute its order. Since the maximum possible number of permutation is $20!$ we'll ask if the order of OT is this value.

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4)(2,3)")
sage: OT=S20.subgroup([R,T])
sage: OT.order()==factorial(20)
True
```

Therefore, *all* permutations of the puzzle pieces are possible. We could have instead asked Sage if OT is the symmetric group S_{20} to achieve the same result.

Sage

```
sage: OT==SymmetricGroup(20)
True
```

Theorem 15.1 (Solvability Criteria for Oval Track puzzle): *For the Oval Track puzzle with* 20 *disks and* $T = (1,4)(2,3)$*, every permutation* $\alpha \in S_{20}$ *is solvable. In other words,* $OT = S_{20}$.

Knowing that all permutation in S_{20} are obtainable is a start, but we would actually like to know how to solve the puzzle from any arrangement of the disks. Moreover, it would be nice to see exactly why Sage is correct in stating $OT = S_{20}$; the algorithms implemented in Sage to do these calculations are beyond the scope of this course.

The theory we have developed provides us with the answer as to why $OT = S_{20}$. In Lecture 13 we found a square commutator that produces a 3-cycle:

$$[R^{-3}, T]^2 = (1,7,4).$$

The puzzle provides us enough flexibility, or "wiggle room", to bring any 3 disks into positions $1,7,4$. See Exercise 3 for some practice in doing this. Therefore we may perform any 3-cycle by conjugation. See Section 15.1.2 for an example. This means we can produce any even permutation of the 20 disks, so $A_{20} < OT$. Also, OT contains an odd permutation: the 20-cycle R. This is enough to conclude that $OT = S_{20}$. See Exercise 6.

This gives a theoretical answer as to *why* every permutation of the disks is possible, but it doesn't provide us with a method, or strategy, to solve any given configuration. We still have some work to do to find out *how* to solve the puzzle.

We begin by looking for a 2-cycle, which we know must exist. Given one 2-cycle we should be able to conjugate it to get all other 2-cycles, given that there seems to be enough "wiggle room".

15.1.1 2-cycles

The most basic combination of moves is TR^{-1}. (Here we use R^{-1} since this brings low numbered disks into the turntable, and this will be handy later when we vary the number of disks used.) This is the product of a 2-cycle and a 17-cycle. In other words, it is a move of order 34. The move $(TR^{-1})^{17}$ has order 2 and is in fact a 2-cycle. Let τ denote this 2-cycle:

$$\tau = (TR^{-1})^{17} = (1,3).$$

Sage

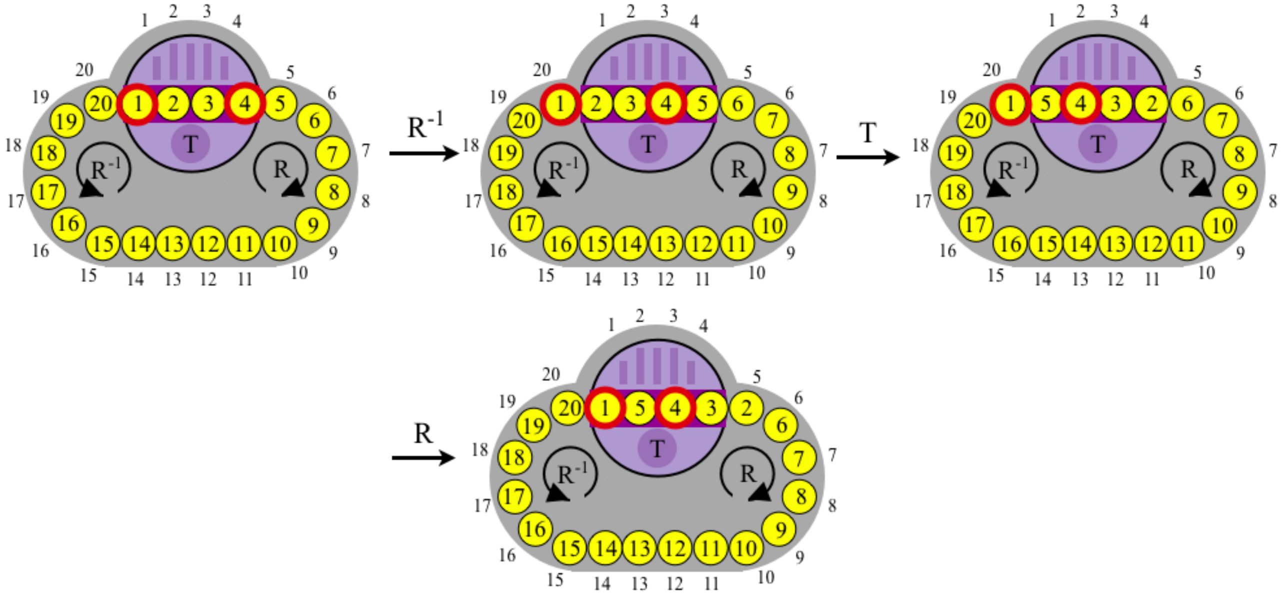
```
sage: T*R^(-1)
(1,3)(4,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5)
sage: (T*R^(-1))^(17)
(1,3)
```

Producing $\tau = (1, 3)$ is a good first step. But it uses quite a few moves: 34 in total. Is it possible to perform a transposition using less moves? Notice that this move sequence sends *every* disk through the turntable, in some sense this sequence of moves is considered “global”. Maybe we could find a “local” move sequence, like the 3-cycle commutator: $[R^{-3}, T]^2 = (1, 7, 4)$, which only puts disks 1 through 7 in the turntable, all other disks are just rocked back-and-forth. Are we able to find a “local” move to produce a 2-cycle?

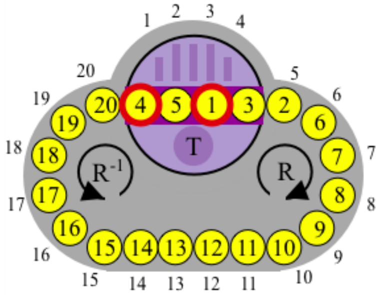
Our theory tells us no! If we think about a local move sequence, it would only use disks 1 through m, all other disks ($m+1$ through 20) would just rock back-and-forth. This means, the same move sequence would produce a 2-cycle on the puzzle with 21 disks for example. Yes, we are changing the puzzle, but this doesn’t affect the 2-cycle, as long as it is “local”. But T is an even permutation, and R would be a 21-cycle, which is even too. Therefore $\langle T, R\rangle$ would only produce even permutations, hence no 2-cycle. Therefore, if we are able to get a 2-cycle in OT it must use a sequence of moves that puts every disk through the turntable at least once. Our move $\tau = (TR^{-1})^{17}$ does this: it sends all disks through the turntable once. This seems to be the best we can do. This is an illustration of the power of the theory we have developed so far. We can answer questions about what we can, and cannot, do with the pieces of the puzzle.

Now that we have one 2-cycle we can conjugate it to get others.

For example, let’s build $(1, 4)$ as a conjugate of $\tau = (1, 3)$. To do this, we will find a sequence of moves that takes 4 to position 3, while at the same time leaving 1 in position 1. The required movement is to push disk 4 one spot to the left (i.e. one spot closer to disk 1). If we rotate the track until 4 is in spot 3, then apply T, we have now moved 4 one spot closer to 1 on the right. Then rotate the track so 1 is back in position 1. You may have noticed we just applied a conjugate to do this: $R^{-1}TR$. See the following diagram.

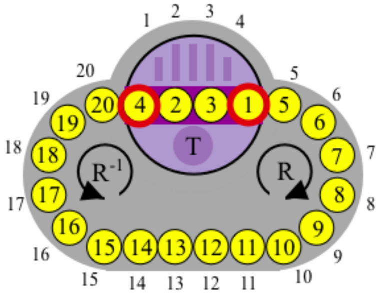


We now swap 1 and 4 using the transposition $\tau = (1, 3)$, which puts the puzzle in following position.



Then undo the set-up moves above to produce $(1,4)$. To summarize, we performed the conjugate

$$(R^{-1}TR)\tau(R^{-1}TR)^{-1} = (1,4).$$



There was nothing special about 1 and 4 in this example. For any two disks a and b we can use turntable moves to bring them closer together, until there is only one disk between them, then we can rotate the track until they are in positions 1 and 3. This results in the set-up move β. Now apply τ, then undo the set-up move: β^{-1}. The result is $\beta\tau\beta^{-1} = (a,b)$. This proves the following.

Theorem 15.2 (2-cycles on Oval Track): *For the Oval Track puzzle with 20 disks and $T = (1,4)(2,3)$, every 2-cycle can be obtained as a conjugate of $(TR^{-1})^{17} = (1,3)$.*

Notice, Theorem 15.1 now follows from this theorem. This provides another proof that $OT = S_{20}$.

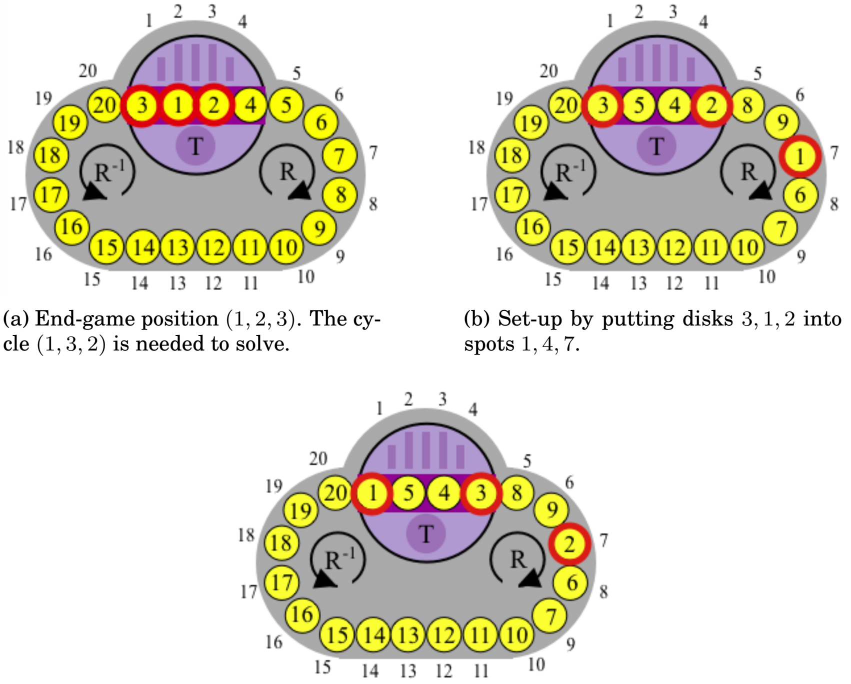
15.1.2 3-cycles

While investigating commutators in Lecture 13 we found a square commutator that produces a 3-cycle:

$$[R^{-3},T]^2 = (1,7,4).$$

Having one 3-cycle is valuable since we can conjugate it to get other 3-cycles. Note, we can't simply assume we can generate all 3-cycles as conjugates since we need to be able to perform a set-up move which takes any 3 disks to spots $1,7,4$. From the example below we'll see that the puzzle provides enough flexibility so that this is always possible.

For example suppose we are solving the puzzle and have brought it to an end-game position $(1,2,3)$. See Figure 15.2a. To solve the puzzle we need to apply the inverse 3-cycle $(1,3,2)$. To accomplish this we will use our fundamental 3-cycle $(1,7,4)$ by first performing a sequence of moves that puts disks 3, 1 and 2 into spots 1, 4 and 7. We will record the sequence of moves as β^{-1}.



(a) End-game position $(1, 2, 3)$. The cycle $(1, 3, 2)$ is needed to solve.

(b) Set-up by putting disks $3, 1, 2$ into spots $1, 4, 7$.

(c) Perform the 3-cycle $(1, 7, 4)$.

Figure 15.2: The steps for performing the 3-cycle $(1, 3, 2)$ as a conjugate of the 3-cycle $(1, 7, 4)$.

Before we start, we look at the current arrangement and make a mental note that “1 chases 3”. By this we mean that disk 1 is to go to the spot where disk 3 is right now. This description will help us decide whether we should perform $(1, 7, 4)$ or $(1, 4, 7)$ at a later time.

It doesn’t matter how you go about getting these three disks into positions 1, 4, and 7. We’ll keep 3 in position 1, move 2 to position 4, and move 1 to position 7. This means we need to space the disks out by adding two spaces between the pairs of disks.

To add spaces we proceed as follows. Move disk 3 to the left, just off the turntable (position 20), and apply T to get some space between it and disks 1 and 2. Now there are two disks between 3 and 2 and disk 1 is just to the right of disk 2. To add space between disk 1 and 2 we move disk 2 to position 20, apply T, which now puts three spaces between 2 and 1, so we close this gap by bringing 1 into position 3 and apply T. Now the three disks are spaced out, and so we just move disk 3 to position 1, and it follows that 2 is now in position 4, and 1 is in position 7. See Figure 15.2b. The move sequence we used to do this was $\beta^{-1} = R^{-1}TR^{-3}TR^{-1}TR^{-5}$.

Now we are ready to apply our fundamental 3-cycle: $(1, 7, 4)$, but we need to know whether we are to apply it or its inverse. This is where our mental note comes in: “1 chases 3”. We need to send disk 1 to where disk 3 is now, this means we should apply $(1, 4, 7)$. The puzzle is now in the position shown in Figure 15.2c.

Finally we undo the set-up move by applying β, and the puzzle is solved.

This example provides the general technique for producing 3-cycles.

Guide for producing a 3-cycle:

Step 1. Pick the three disks you wish to cycle: (a, b, c). Make a mental note that “a chases b”.

Step 2. Move the disks to positions $1, 4, 7$, in any way whatsoever. Call this move β^{-1}.

Step 3. Apply the fundamental 3-cycle $(1, 4, 7)$ or its inverse $(1, 7, 4)$, depending on locations of a and

b.

Step 4. Undo the set-up move by applying β. The result is the 3-cycle (a, b, c).

15.1.3 Strategy for Solution

We are now ready to describe a strategy for solving the Oval Track puzzle. Since we can perform any 2-cycle we already have a method at hand. However, the fundamental 2-cycle is $(TR^{-1})^{17} = (1, 3)$ is 34 moves long, and any other 2-cycle obtained by conjugation will use more moves. So solving the puzzle by swapping two pieces at a time is very inefficient.

Similarly, we can create any 3-cycle by conjugating the fundamental 3-cycle $[R^{-3}, T]^2 = (1, 7, 4)$. But again this will result in fairly long move sequences.

Instead, we will just approach the puzzle by first setting pieces 20 through 5 in order, which is fairly straightforward since there is enough "wiggle room" to move things around. This brings the puzzle to its end-game position, that is, a position where only disks $1, 2, 3, 4$ are permuted. It is at this point where 2-cycles and 3-cycles will be useful. Moreover, we will try to use 3-cycles since the move sequence is significantly shorter, but if forced we may need to use a 2-cycle, which we have ready and waiting.

Will we ever be forced to use a 2-cycle? In the end game all permutations in S_4 are possible. For example, we may be faced with an end-game configuration which is an odd permutation. Since 3-cycles are even we won't be able to solve it using 3-cycles alone, we will be forced to use at least one 2-cycle.

Guide to solve the puzzle:

(a) Starting with disk 20 put disks 20 through 5 in numerical order.

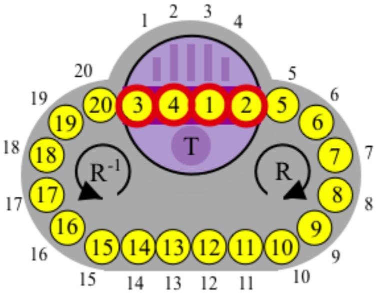
(b) The permutation of the final 4 disks is either even or odd. This is the *end-game* phase.

   (a) If the permutation is even, express it is either a 3-cycle or a product of two 2-cycles.
   3-**cycle**: Use a conjugate of the fundamental 3-cycle $[R^{-3}, T]^2 = (1, 7, 4)$ to solve it.
   two 2-**cycles**: Check whether it is $(1, 4)(2, 3)$, if it is apply T and you're done. Otherwise, express it as the product of 3-cycles (recall $(a, b)(c, d) = (a, c, d)(a, c, b)$), and use conjugates of the fundamental 3-cycle $[R^{-3}, T]^2 = (1, 7, 4)$ to solve it.

   (b) If the permutation is odd, it is either a 2-cycle or a 4-cycle.
   2-**cycle**: Use a conjugate of the fundamental 2-cycle $(TR^{-1})^{17} = (1, 3)$ to solve it.
   4-**cycle**: First check whether apply T reduces the 4-cycle to a 2-cycle. Otherwise, there is a 3-cycle that does. Once you have reduced it to a 2-cycle use a conjugate of the fundamental 2-cycle $(TR^{-1})^{17} = (1, 3)$ to solve it.

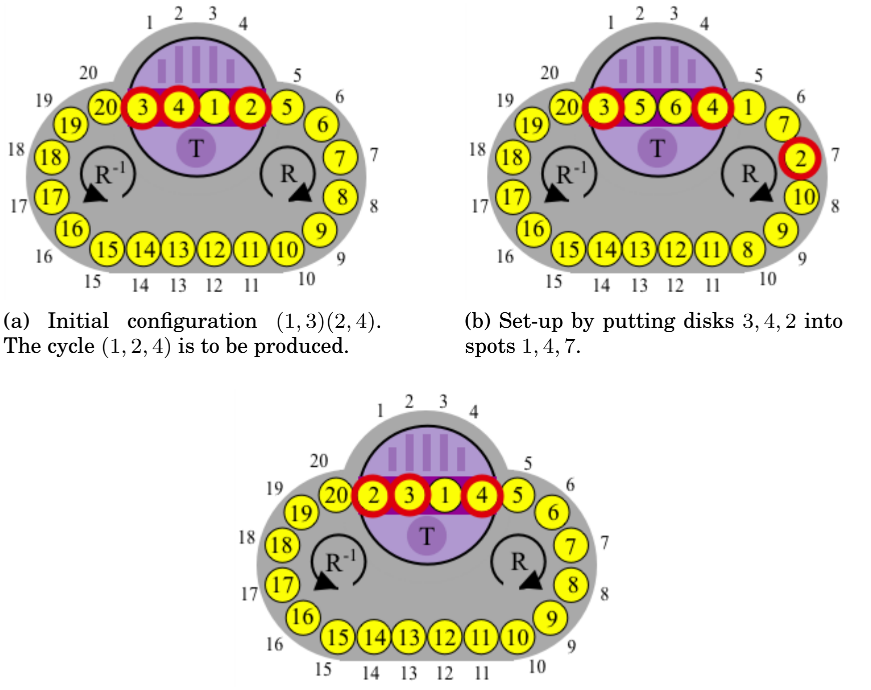
Let's now practice a few end-game configuations.

Example 15.1: Solve the end-game configuration $(1, 3)(2, 4)$.



To solve the puzzle we need to produce the inverse permutation, which is just itself, $(1,3)(2,4)$. Since $(1,3)(2,4)$ is an even permutation we can write it as a product of 3-cycles: $(1,2,4)(1,2,3)$. We focus on constructing each 3-cycle as a conjugate of the fundamental 3-cycle.

$\mathbf{(1,2,4)}$: This 3-cycle involves disks $3,4,2$ and results in putting 4 in its home position. See Figure 15.3a.



(a) Initial configuration $(1,3)(2,4)$. The cycle $(1,2,4)$ is to be produced.

(b) Set-up by putting disks $3,4,2$ into spots $1,4,7$.

(c) Perform the 3-cycle $(1,4,7)$, then undo the set-up move.

Figure 15.3: The steps for performing the 3-cycle $(1,2,4)$ as a conjugate of the 3-cycle $(1,4,7)$.

The direction we want to cycle these disks is summarized by "3 chases 4". Apply the strategy of spacing out the disks by making sure there are two disks between the middle and each outer disk. A move sequence that does this is

$$\beta^{-1} = R^{-1}TR^{-1}TR^{-3}TR^{-1}TRTR^{5}.$$

The puzzle will be as shown in Figure 15.3b.

Recalling that 3 chases 4, the fundamental 3-cycle we should apply is $[R^{-3},T]^{-2} = (1,4,7)$. Then applying β to undo the set-up, we end up with the puzzle in the configuration shown in Figure 15.3c.

$\mathbf{(1,2,3)}$: This 3-cycle involves disks $2,3,1$ and the direction we want to cycle these disks is summarized by "2 chases 3". See Figure 15.4a.

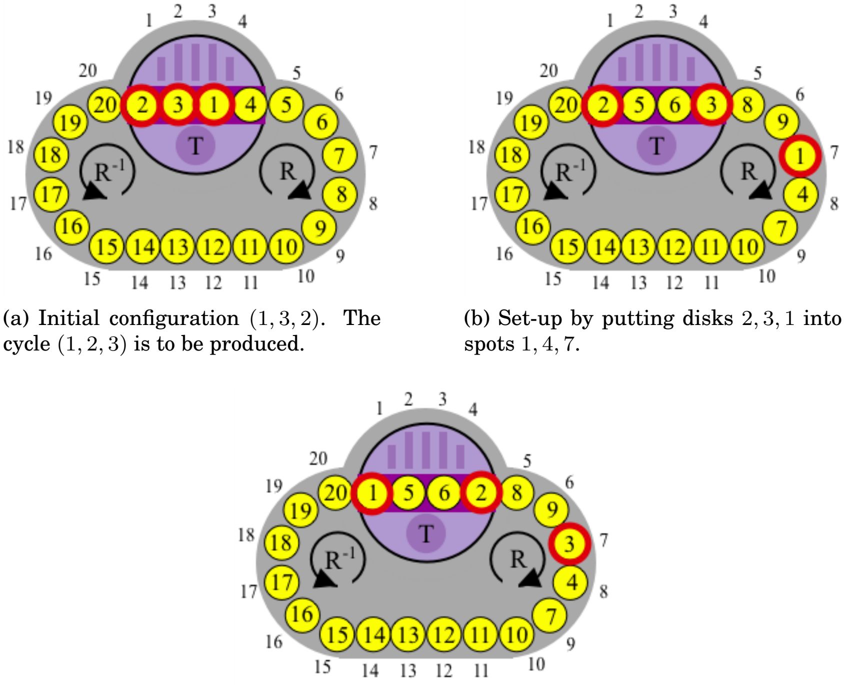
We can space out the disks, making sure there are two disks between the middle and each outer disk, by using the move sequence

$$\delta^{-1} = R^{-1}TR^{-1}TR^{-2}TR^{-1}TR^{5}.$$

The resulting position is shown in Figure 15.4b.

Since 2 is to chase 3, the fundamental 3-cycle we should apply is $[R^{-3},T]^{-2} = (1,4,7)$. See Figure 15.4c. Applying δ to undo the set-up move solves the puzzle.

In the next example we consider the case when the end-game permutation is a 2-cycle.



(a) Initial configuration $(1,3,2)$. The cycle $(1,2,3)$ is to be produced.

(b) Set-up by putting disks $2,3,1$ into spots $1,4,7$.

(c) Perform the 3-cycle $(1,4,7)$.

Figure 15.4: The steps for performing the 3-cycle $(1,2,3)$ as a conjugate of the 3-cycle $(1,4,7)$.

Example 15.2: Solve the end-game configuration $(1,2)$. See Figure 15.5a.

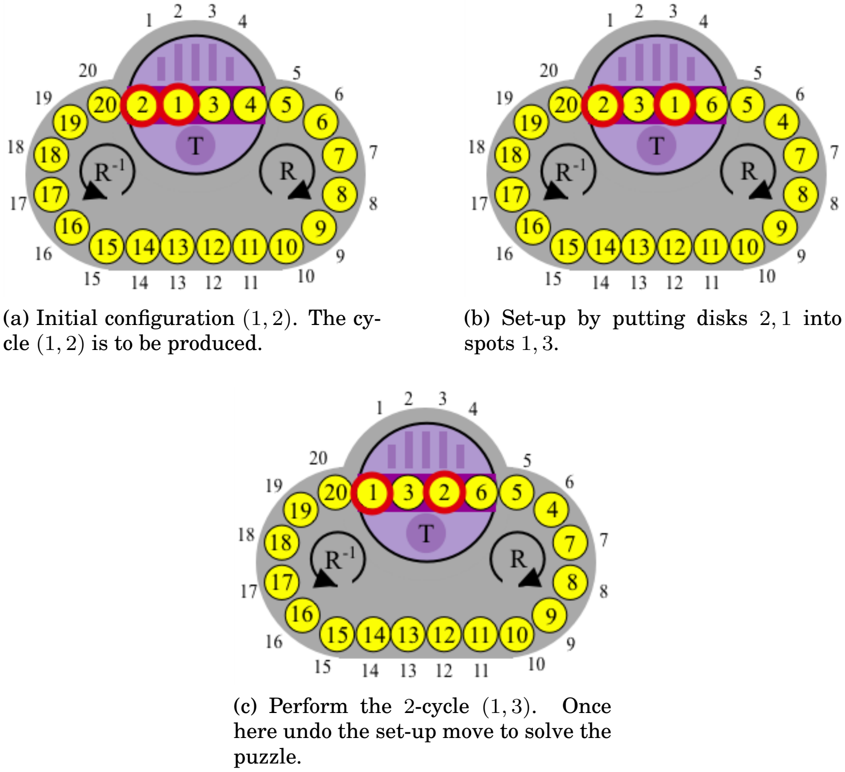
To solve the puzzle we need to produce the inverse permutation, which is just itself, $(1,2)$. Since $(1,2)$ is a 2-cycle we construct it as a conjugate of the fundamental 2-cycle $(TR^{-1})^{17} = (1,3)$. Apply a set-up move which leaves 2 in spot 1, and moves 1 to spot 3. One such move sequence is $\beta^{-1} = R^{-1}TR^{-1}TRTR$. Recall that to do this you just want to insert two disks between disks 2 and 1. The puzzle should now look like Figure 15.5b. Apply the fundamental 2-cycle $(TR^{-1})^{17} = (1,3)$, which results in swapping disks 2 and 1. This is shown in Figure 15.5c. Undoing the set-up move by applying β solves the puzzle.

The end-game permutation could be a 4-cycle, which is an odd permutation. If we are lucky a move T will take it to a transposition as the next example illustrates.

Example 15.3: Solve the end-game configuration $(1,3,2,4)$. See Figure 15.6a.

Every disk is out of place, but disk 4 can be moved to spot 4 by move T. This also brings disk 3 home as well. The permutation of the puzzle pieces is now $(1,2)$ (see Figure 15.6b) which we already solved in the last example.

Contrary to the last example, it may happen the an end-game 4-cycle cannot be immediately converted into a 2-cycle by performing move T. In this case there will always be a 3-cycle that does. Just use a 3-cycle to send any piece home, it follows that some other piece must also be sent home as well. The reason for this is the product of an odd permutation and an even permutation is odd, and the only odd permutations on 3 objects are transpositions. See Exercise 5 for one such end-game.

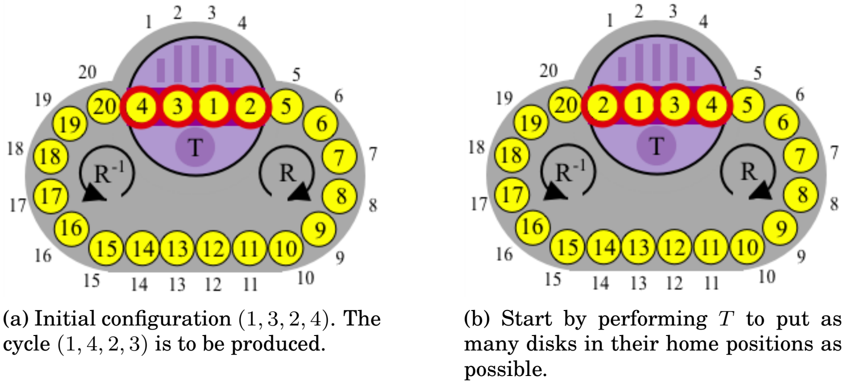


(a) Initial configuration $(1, 2)$. The cycle $(1, 2)$ is to be produced.

(b) Set-up by putting disks $2, 1$ into spots $1, 3$.

(c) Perform the 2-cycle $(1, 3)$. Once here undo the set-up move to solve the puzzle.

Figure 15.5: The steps for performing the 2-cycle $(1, 2)$ as a conjugate of the 2-cycle $(1, 3)$.



(a) Initial configuration $(1, 3, 2, 4)$. The cycle $(1, 4, 2, 3)$ is to be produced.

(b) Start by performing T to put as many disks in their home positions as possible.

Figure 15.6: The 4-cycle $(1, 3, 2, 4)$ is only one move T away from the 2-cycle $(1, 2)$.

15.1.4 Changing the number of disks

What happens if we change the number of disks in the puzzle. For example, suppose we used only 19 disks instead of 20. Would we expect our results to be the same. For example, does Theorem 15.1 remain true for 19 disks? Let's ask Sage .

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,19)")
sage: T=S20("(1,4)(2,3)")
sage: OT19=S20.subgroup([R,T])
```

```
sage: OT19.order == factorial(19)
False
```

Let OT_{19} be the Oval Track group on 19 disks. Then we determined that $|OT_{19}| \neq 19!$, so OT_{19} does not contain every permutation of the 19 disks. This shouldn't come as a surprise though, since the rotation move R is a 19-cycle, which is even, and the turntable move T is also even. Therefore we can only generate even permutations, so at best we could get the group of all even permutations A_{19}. Let's see if we get all of A_{19}.

Sage

```
sage: OT19 == AlternatingGroup(19)
True
```

We do! This means that for the Oval Track puzzle on 19 disks the solvable permutations are precisely the even permutations.

What happened to our fundamental 2-cycle $(TR^{-1})^{17}$? It was built from TR^{-1} so let's see what TR^{-1} is now.

Sage

```
sage: T*R^(-1)
(1,3)(4,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5)
```

It is now a product of a 2-cycle and a 16-cycle. Unlike the 20 disk case, there is no way to take a power of this to kill-off the 16-cycle and leave the 2-cycle alone. However, we still have our fundamental 3-cycle: $[R^{-3}, T]^2 = (1, 7, 4)$ so we can use conjugates of this to solve the end-game of this puzzle.

What about changing the number of disks even further? Let n be the number of disks, and let OT_n be the Oval Track group on n disks. Notice the move R, which is an n-cycle, will be even if and only if n is odd. Therefore OT_n will contain only even permutations: $OT_n \leq A_n$. On the other hand, if n is even then R is an odd permutation, so OT_n will contain some odd permutations. The questions are then: (i) for n odd is $OT_n = A_n$, and (ii) for n-even is $OT_n = S_n$?

We can use Sage to help us answer these questions. Here we consider the number of disks $4 \leq n \leq 20$.

Sage

```
sage: for n in (4..20):
sage:     Rn=S20([tuple(range(1,n+1))])    # creates n-cycle (1,2,3,...,n)
sage:     OTn=S20.subgroup([Rn,T])         # creates OTn: the Oval Track group on n disks
sage:     if is_even(n):
sage:         print n, OTn==SymmetricGroup(n)  #check if OTn is the symmetric group
sage:     else:
sage:         print n, OTn==AlternatingGroup(n) #check if OTn is the alternating group
4 False
5 False
6 True
7 True
8 True
9 True
10 True
11 True
12 True
13 True
14 True
15 True
16 True
17 True
```

```
18 True
19 True
20 True
```

Therefore, for $n \geq 6$ the answers to our questions are: yes. However, for small values of n the answer is: no. It seems like there just isn't enough "wiggle room" to get all the permutations when there is a small number of disks.

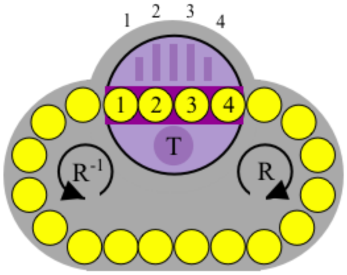
Let's investigate this further.

If $n \geq 6$, the product TR^{-1} consists of a 2-cycle and an $(n-3)$-cycle: disk 2 remains fixed, disks 1 and 3 are swapped, and the remaining $n-3$ disks are cycled to the left around the track. If n is even, then $n-3$ is odd so $(TR^{-1})^{n-3}$ is a 2-cycle $(1,3)$. Having this 2-cycle, and using conjugation, indicates why $OT_n = S_n$ when n is even.

If $n \geq 7$ we still have the fundamental 3-cycle $[R^{-3}, T]^2 = (1,7,4)$, so we can conjugate it to get other 3-cycles. This indicates why $OT_n = A_n$ when n is odd.

The remaining cases are $n = 4, 5$.

$\boldsymbol{n = 4}$: We can view this puzzle as in the diagram, where only the labeled disks are in play and they are free to move around the track by the rotation $R = (1,2,3,4)$.

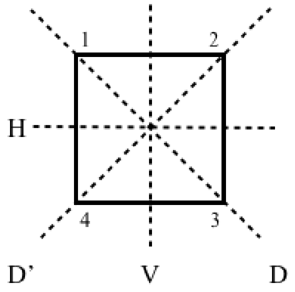


We use Sage to work out the order of OT_4.

Sage

```
sage: S4=SymmetricGroup(4)
sage: R=S4("(1,2,3,4)")
sage: T=S4("(1,4)(2,3)")
sage: OT4=S4.subgroup([R,T])
sage: OT4.order()
8
sage: OT4.list()
[(), (2,4), (1,2)(3,4), (1,2,3,4), (1,3), (1,3)(2,4), (1,4,3,2), (1,4)(2,3)]
```

It is a group of order 8, and these elements look very familiar. They remind us of another group of order 8 we know, the dihedral group D_4, which is the group of symmetries of a square. If we label the vertices of the square by $1, 2, 3, 4$, then the group of symmetries D_4 can be viewed as a subgroup of S_4.



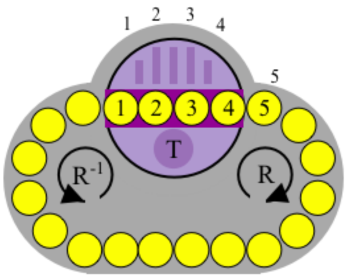
Sage
```
sage: D4=DihedralGroup(4)
sage: D4.list()
[(), (2,4), (1,2)(3,4), (1,2,3,4), (1,3), (1,3)(2,4), (1,4,3,2),(1,4)(2,3)]
```

A rotation R of the pieces along the track, corresponds to a rotation of the square. A turntable move correspond to a reflection about the horizontal axis.

Since OT_4 and D_4 are essentially the same group, it is just the context that is different, we say these groups are **isomorphic** and write

$$OT_4 \approx D4.$$

$n = 5$: We can view this puzzle as in the diagram, where only the labeled disks are in play and they are free to move around the track by rotation the $R = (1, 2, 3, 4, 5)$.

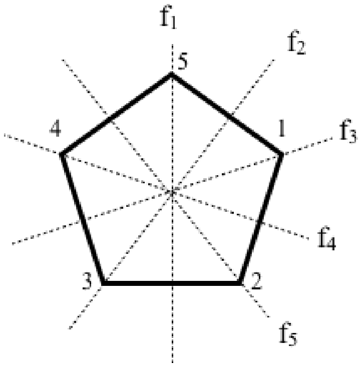


Sage
```
sage: S5=SymmetricGroup(5)
sage: R=S5("(1,2,3,4,5)")
sage: T=S5("(1,4)(2,3)")
sage: OT5=S5.subgroup([R,T])
sage: OT5.order()
10
```

Based on our experience with $n = 4$, and the fact that the dihedral group of a regular pentagon has order 10, we may suspect that $OT_5 \approx D_5$. Checking with Sage we see this is indeed the case.

Sage
```
sage: OT5==DihedralGroup(5)
True
```

Since spot 5 is not on the turntable, move T is analogous to reflection f_1 of the pentagon in the digram below. This analogy indicates that the symmetries of the pentagon are generated by a clockwise rotation and the reflection f_1.



We summarize our results in the following theorem.

Theorem 15.3: *On the Oval Track puzzle with* $T = (1,4)(2,3)$ *, any scrambling can be solved if the number of disks* n *is even and* $n \geq 6$*. If* $n \geq 7$ *and odd then every even scrambling can be solved. Under these latter conditions, odd permutations can be brought down to a single transposition, but cannot be completely solved. In particular, if* OT_n *denotes the group of permutations achievable by the Oval Track puzzle with* n *disks then:*

$OT_4 \approx D_4$,

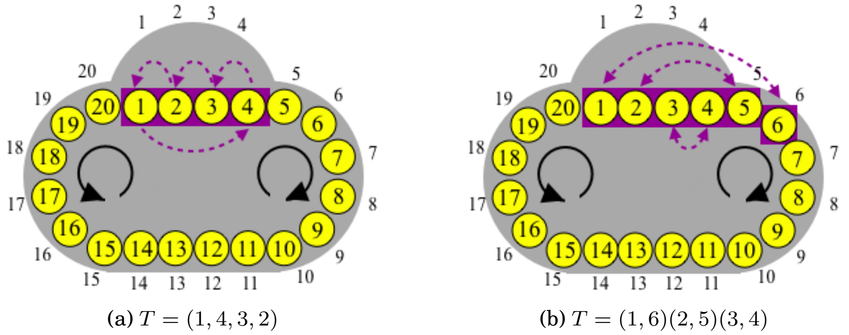
$OT_5 \approx D_5$,

$OT_n \approx S_n$ *if* $n \geq 6$ *and even,*

$OT_n \approx A_n$ *if* $n \geq 6$ *and odd,*

15.2 Variations of the Oval Track T move

Variations of the Oval Track puzzle can be created by changing the turntable move T. Figure 15.7 shows two different variations.



(a) $T = (1,4,3,2)$

(b) $T = (1,6)(2,5)(3,4)$

Figure 15.7: Some variation of the turntable move T for the Oval Track puzzle.

We can use Sage to investigate the groups associated with these variations.

```
sage: S20=SymmetricGroup(20)
```

```
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,4,2,3)")
sage: OTv1=S20.subgroup([R,T])
sage: OTv1==SymmetricGroup(20)
True
```

Sage

```
sage: S20=SymmetricGroup(20)
sage: R=S20("(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)")
sage: T=S20("(1,6)(2,5)(3,4)")
sage: OTv2=S20.subgroup([R,T])
sage: OTv2==SymmetricGroup(20)
True
```

Therefore, on both these puzzles, all permutations of the pieces are possible.

Coming up with a strategy to solve these puzzles is similar to how we approached the original Oval Track puzzle. Use commutators to create moves, and conjugates to modify them. Try to find a fundamental 3-cycle or 2-cycle.

For the first variation, where $T = (1,4,3,2)$, we have commutators $[R^{-1}, T] = (1,2,5)$, and $[T^{-1}, R^{-1}] = (1,5,4)$. The product of these two is

$$[R^{-1}, T][T^{-1}, R^{-1}] = (1,2,4).$$

What is interesting about this is that combining this with T gives a 2-cycle:

$$[R^{-1}, T][T^{-1}, R^{-1}]T = (2,3).$$

We know how important having a 2-cycle is for solvability.

15.3 Exercises

1. **Getting to the end-game position.** Go to Jaaps Puzzle page [13] and play with the javascript "Top Spin" puzzle. Mix up the disks and try to restore disks 20 through 5. That is, reduce the puzzle down to the end-game position. Do this a number of times until you are confident that getting to the end-game position is fairly straightforward. Don't worry about solving the end-game just yet.

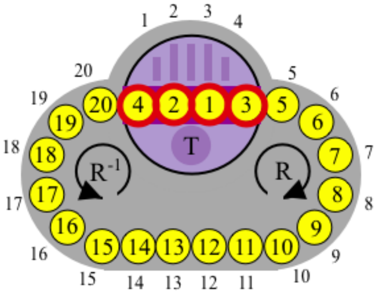
2. **2-cycles on OT with $T = (1,4)(2,3)$.** For each of the following 2 cycles, find a conjugate of $\tau = (1,3)$ which produces the 2-cycle. That is, find a sequence of moves β^{-1} so the $\beta^{-1}\tau\beta$ produces the desired 2-cycle.

   (a) $(1,9)$ (b) $(1,2)$ (c) $(3,14)$ (d) $(2,11)$

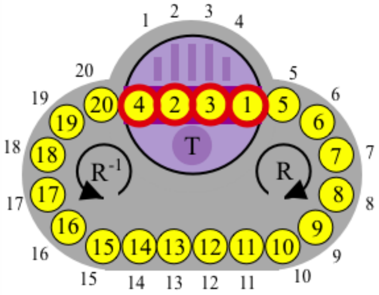
3. **3-cycles on OT with $T = (1,4)(2,3)$.** For each of the following 3 cycles, find a conjugate of the fundamental 3 cycle $\sigma = (1,4,7)$, or its inverse σ^{-1} which produces the 3-cycle. That is, find a sequence of moves β^{-1} so the $\beta^{-1}\sigma\beta$ produces the desired 3-cycle.

   (a) $(1,4,3)$ (b) $(1,3,4)$ (c) $(2,3,4)$

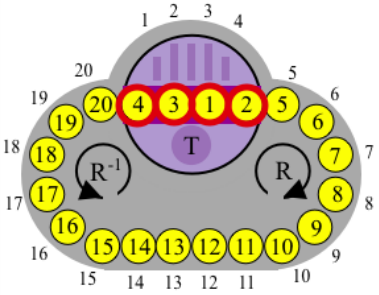
4. There are six end-game configurations shown below. (a) Write out each one in cycle notation. (b) Plan a strategy for solving the end-game. (c) Implement your strategy and solve each of the puzzles. You may find it useful to use the virtual puzzles on Jaaps Puzzle page [13] to try out your move sequences.



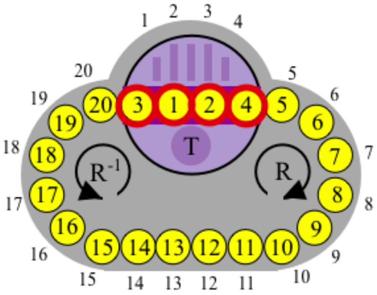
(a)



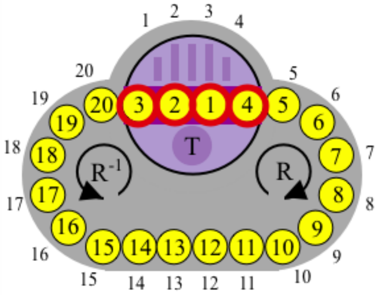
(b)



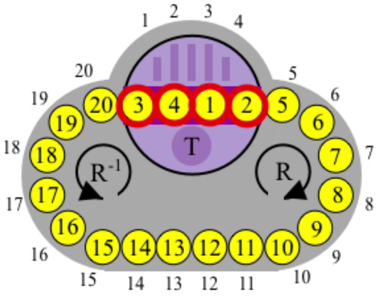
(c)



(e)

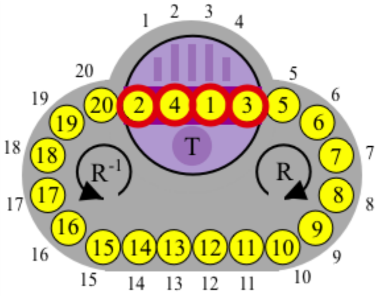


(f)



(g)

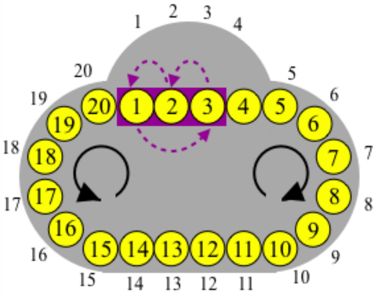
5. Solve the end-game configuration $(1, 3, 4, 2)$, which is shown in the diagram.



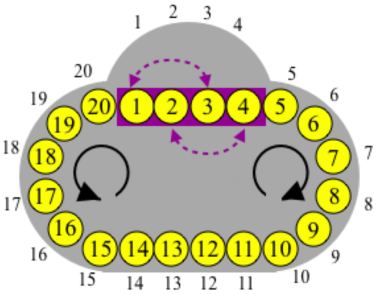
6. **Getting all permutations from one odd, and A_n.** Let $G < S_n$ be a group of permutation which contains all even permutations (i.e. $A_n < G$). and has at least one odd permutation $\beta \in G$. Show that $G = S_n$.

   (Hint: We already know the set of odd permutation O_n is the same size as the set of even permutations A_n (see Theorem 8.1). It suffices to show we can get all the elements of O_n from A_n and β. Show $O_n = \beta A_n := \{\beta\alpha \mid \alpha \in A_n\}$.)

7. Consider the variation $T = (1, 3, 2)$ of the turntable move on the Oval Track puzzle with 20 disks. Are all permutations of the puzzle pieces possible?



8. Consider the variation $T = (1,3)(2,4)$ of the turntable move on the Oval Track puzzle with 20 disks. Are all permutations of the puzzle pieces possible?



Lecture 16

Mastering the Hungarian Rings Puzzle

In this lecture we give a thorough analysis of the Hungarian Rings puzzle, both the coloured and the numbered versions.

16.1 Hungarian Rings - Numbered version

In this section we focus on the Hungarian Rings puzzle as shown in Figure 16.1. It seems reasonable that the numbered version is more difficult to solve than the coloured version. This is because in the coloured version has only 4 distinct disks, but the numbered version has 38 distinct disks. Even though it is more difficult we will start with the numbered version. In Section 16.4 we will apply our new-found knowledge to the coloured version and describe a simple and elegant solution.

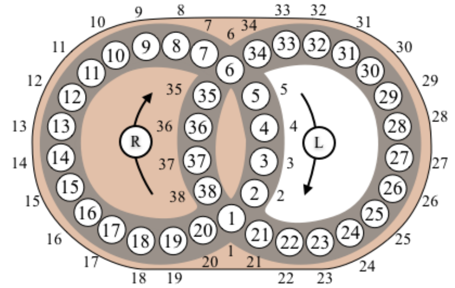


Figure 16.1: Hungarian Rings puzzle - numbered version.

The two basic moves of the Hungarian Rings puzzle are L, and R, where L denotes a clockwise rotation of disks around left ring, where each disk moves one space, and R denotes a clockwise rotation of numbers around the right ring.

The permutation corresponding to the legal moves R and L are as follows:

$$L = (1, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2)$$
$$R = (1, 38, 37, 36, 35, 6, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21)$$

and the Hungarian Rings puzzle group is $HR = \langle L, R\rangle$.

Note that L^{-1} and R^{-1} represents a counterclockwise rotation of the disks around the respective rings.

Let's get right down to business and find out which permutations of the 38 disks are possible. We can set-up the corresponding puzzle group HR in Sage and compute its order. Since the maximum possible number of permutation is 38! we'll ask if the order of HR is this value.

Sage

```
sage: S38=SymmetricGroup(38)
sage: L=S38("(1,20,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2)")
sage: R=S38("(1,38,37,36,35,6,34,33,32,31,30,29,28,27,26,25,24,23,22,21)")
sage: HR=S38.subgroup([L,R])
sage: HR.order()==factorial(38)
True
```

Therefore, *all* possible permutations of the puzzle pieces are possible. We could have instead asked if HR is the symmetric group S_{38} to achieve the same result.

Sage

```
sage: HR==SymmetricGroup(38)
True
```

Theorem 16.1 (Solvability Criteria for Hungarian Rings puzzle): *For the Hungarian Rings puzzle every permutation of the 38 pieces is possible. In other words, $HR = S_{38}$.*

Much like the Oval Track puzzle we can see theoretically why $HR = S_{38}$. In Lecture 14 we saw that we could produce a 3-cycle as a compound conjugate:

$$[[L^5, R^5], R^{-1}LR] = (6, 11, 12).$$

There is enough "wiggle room" on the puzzle to bring any three disks into spots $6, 11, 12$, so we can perform any 3-cycle by conjugating this one. Therefore, we can perform any even permutation of the puzzle pieces. The move L is a 20-cycle, which is odd. This means HR contains A_{38} and at least one odd permutation. Therefore it must contain all of S_{38}. This is similar to the argument used to show $OT = S_n$ when the number of disks $n \geq 6$ is even.

Knowing that all permutations in S_{38} are obtainable is a start, but we actually would like to know how to solve the puzzle from any arrangement of the disks. As with the Oval Track puzzle, moving the first few disks home is straightforward, it is the end-game where we need theory-based strategies.

16.1.1 Start-game: Solve the first 20 disks

There is enough flexibility in the puzzle to solve disks 7 through 16 on the left ring, and disks 21 through 30 on the right ring. You may be able to get a couple more disks in place, such as 17 and 31.

Using general heuristics you may be able to get a few more disks into place. Once you are at the point where you think general heuristics cannot take you any further you are at the end-game. You will probably have 20 to 23 disks in their proper position. This leaves 15 to 18 disks we still need to solve.

16.1.2 End-game: A strategy

This puzzle has a rather large end-game, as compared to the Oval Track puzzle. Once we have made it to this point we express the remaining permutation in disjoint cycle form. It will possibly involve 2-cycles, 3-cycles, 4-cycles, 5-cycles, and perhaps longer cycles.

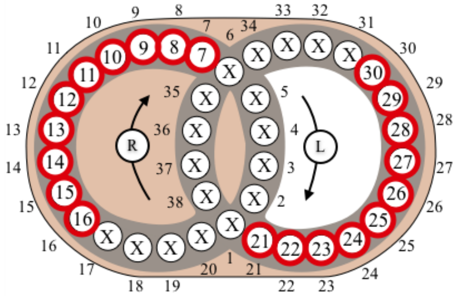


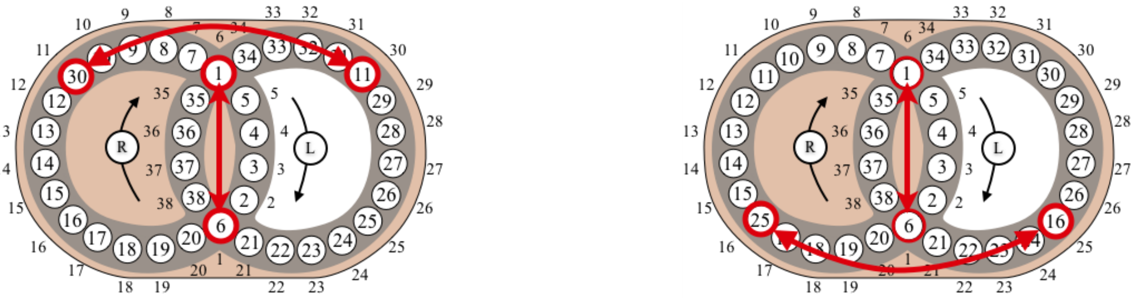
Figure 16.2: Start-game: begin by putting disks 7 – 16 and 21 – 30 in place.

If we know some fundamental cycles of these lengths then we have a strategy to solve: just solve one cycle at a time. In the next section we go about building fundamental cycles.

Before doing this. let's recall the fundamental commutators (see Figure 16.3):

$$[L^5, R^{-5}] = (1,6)(11,30), \quad \text{and} \quad [L^{-5}, R^5] = (1,6)(16,25).$$

These will come in handy in the following sections. We'll come back to the end game strategy in Section 16.3.



(a) $L^5R^{-5}L^{-5}R^5 = (1,6)(11,30)$

(b) $L^{-5}R^5L^5R^{-5} = (1,6)(16,25)$

Figure 16.3: Basic commutators on the Hungarian Rings puzzle

16.2 Building Small Cycles: Tools for Our End-Game Toolbox

16.2.1 5-cycles

Starting with the intersection spots 1 and 6, there is a collection of 6 spots that are nicely spaced around the puzzle: each one five away from the next one. The locations of these spots are 1, 6, 11, 16, 24, 30. With this observation, there is a nice 8-move sequence to creates a 5-cycle:

$$\sigma_5 = (L^5R^5)^4 = (1,25,30,11,16).$$

Conjugation of this 5-cycle will come in useful when we need to deal with long cycles in the end-game.

16.2.2 4-cycles

Think back to the Oval Track puzzle and how we produced a 2-cycle. We had to send every disk through the turntable. It was theoretically impossible to produce a 2-cycle without doing this. The reason, as we

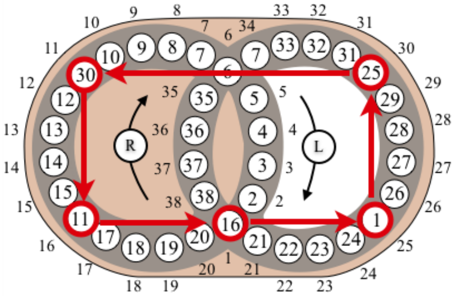


Figure 16.4: Fundamental 5-cycle: $\sigma_5 = (L^5R^5)^4 = (1, 25, 30, 11, 16)$.

discussed in Lecture 14, was that if one or more disks never passed through the turntable then it would be possible to do the same thing on the puzzle with 21 disks. But this puzzle doesn't have a 2-cycle since the basic moves are even. A similar argument would show that *every* odd permutation on the Oval Track puzzle must come from a sequence of moves that push every piece through the turntable.

There is a similar theoretical result for the Hungarian Rings puzzle.

Theorem 16.2: *On the Hungarian Rings puzzle, suppose there is a sequence of moves that produces an odd permutation β, which returns at least one disk on each ring to its home position. Then during the process, each piece r on the right ring where $\beta(r) = r$ must have been temporarily moved to the left ring, or each piece ℓ on the left ring where $\beta(\ell) = \ell$ must have been temporarily moved to the right ring. In other words, every piece on one of the rings that β keeps at home would have been temporarily sent to the other ring.*

Proof: To see why this is true, suppose β is an odd permutation that keeps a disk r on the right ring at home and keeps a disk ℓ on the left ring at home (i.e. $\beta(r) = r$ and $\beta(\ell) = \ell$), and suppose during the entire process it keeps r on the right ring, and ℓ on the left ring. Without loss of generality, we can assume $\beta = L^{m_1}R^{k_1}L^{m_2}R^{k_2}\cdots L^{m_\ell}R^{k_\ell}$, for integers m_i and k_j, in which some could be 0.

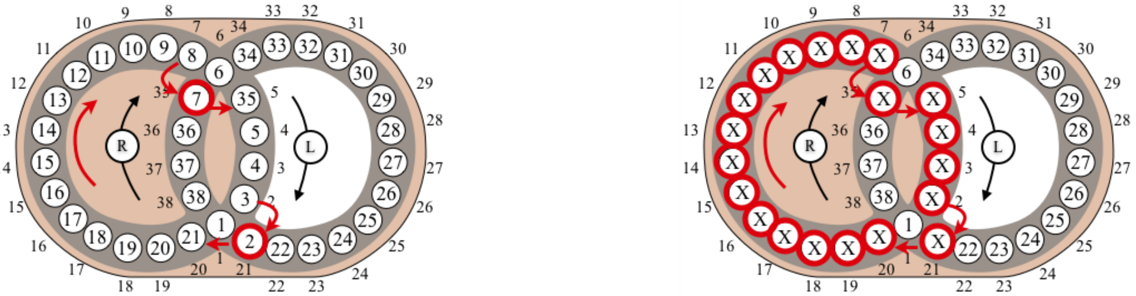
Since r is never moved to the left ring, the only moves that affect it are the moves R^{k_i}, so the overall effect of β on r is the same as that of $R^{k_1}R^{k_2}\cdots R^{k_\ell} = R^{k_1+k_2+\cdots+k_\ell}$, which turns the right ring $k_1 + k_2 + \ldots + k_\ell$ positions. Since r is returned home then $k_1 + k_2 + \cdots + k_\ell$ must be divisible by 20, and hence the right ring moves contribute an even permutation to the process.

Similarly, by considering piece ℓ on the left ring, $m_1 + m_2 + \cdots m_\ell$ is divisible by 20, and so the left ring moves contribute an even permutation to the process. Therefore β must be even. A contradiction. □

Theorem 16.2 provides insight as to how to construct a 4-cycle, or any odd permutation for that matter: for every disk on one ring that is to remain fixed by the permutation, we need to temporarily move it to the other ring. Let's try to do this with the disks on the left ring. The reason we use the left ring is purely aesthetic: the left ring consists of the numbers are 1 through 20 which are easy to remember.

First let's draw our attention to disks 35 and 21 in the solved state of the puzzle. See Figure 16.6a. We'd like to consider a move which only affects these disks in the right ring. Recall that our goal is to temporarily move every disk in the left ring to the right ring, as this is necessary if we wish to construct an odd permutation. To simplify what could potentially be a complicated set of moves, we would like to minimize the number of pieces that are moved on the right ring. We will only try to use move-sequences that affect positions 35 and 21 of the right ring, and these will be the positions where the left ring pieces temporarily visit.

The conjugate RLR^{-1} is a move that only affects positions 35 and 21 in the right ring, so let's begin with that move. It temporarily moves 1 and 6 off the left ring via move R, puts them in the holding spots (positions 38 and 34, respectively), after move L is applied then R^{-1} moves them back on the left ring to where they started. It also moves disks 2 and 7 off the left ring, and leaves them in our holding spots (positions 21 and 35, respectively) on the right ring. See Figure 16.5a. If we do this move again, it will put 2 back on the left ring, but it will be on the opposite side of 1. It also moves 1 and 6 off and on again. Repeated applications would keep moving 1 and 6 off and on the right ring, while at the same time moving another two disks off, then eventually back on. Figure 16.5b shows the result of repeated application of RLR^{-1}.



(a) The affect of applying RLR^{-1} once.

(b) Each application of RLR^{-1} moves the disks marked X one unit along the path.

Figure 16.5: The result of applying move RLR^{-1} once, and repeatedly.

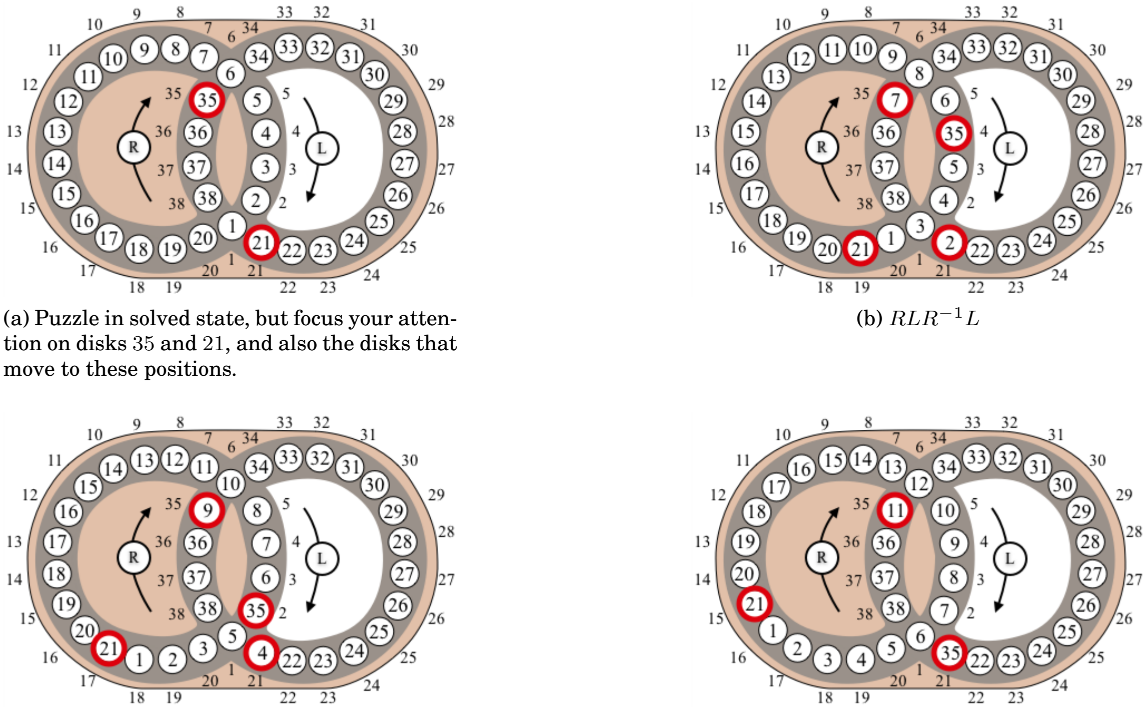
This would be a very slow process, to move every tile off the left ring then back on again, not to mention at some point we would need to do something to control which odd permutation we construct. Instead, it would be nice to move as many numbers off and on the left ring as possible, in a minimum number of moves, while at the same time keeping as many disks as we can in numerical order. To achieve this, we consider RLR^{-1} as the first move, then we advance the numbers on the left ring, before applying RLR^{-1} again, this would put two new numbers in positions 1 and 6 which would then be ready to be moved off and back on the left ring with the next application of RLR^{-1}. In other words, let's consider the move sequence $RLR^{-1}L$. The result of this move is shown in Figure 16.6b (in the figure we've drawn our attention to positions and disks 21 and 35).

There are a few things that we should note about the move $RLR^{-1}L$:

- All disks on the right ring were unaffected, except for disk 35 and 21.
- The disks in positions 7 and 2 were moved to storage on the right ring. And disks in positions 35 and 21 moved to take their place on the left ring.
- The disks in positions 1 and 6 were temporarily moved to positions 38 and 34 on the right ring, and then move back to the left ring, ending up on positions 20 and 5, respectively. That is, they moved only position clockwise around the ring.
- All other disks on the left ring advanced two positions clockwise around the ring.

Repeated application of $RLR^{-1}L$ is shown in Figure 16.6. A summary of which disks are moved off the left ring, then back on again, by repeated application of $RLR^{-1}L$ is given in Table 16.1. It is important to notice the change that was made by $(RLR^{-1}L)^3$, so compare Figure 16.6a to 16.6d. Disks that started in positions 1 to 10, are now back in their natural order after having been moved temporarily to the right ring. Disks that started in positions 12 through 20 are still in their natural order (they weren't moved to the right ring).

If we continue to repeat the process then we would disturb the natural order of disk 1 to 10. Instead, we first rotate the left ring so that disks 1 through 10 are out of the way (apply L^5), then we apply the



(a) Puzzle in solved state, but focus your attention on disks 35 and 21, and also the disks that move to these positions.

(b) $RLR^{-1}L$

(c) $(RLR^{-1}L)^2$

(d) $(RLR^{-1}L)^3$

Figure 16.6: The set-up moves for creating the 4-cycle $(1, 35, 11, 21)$.

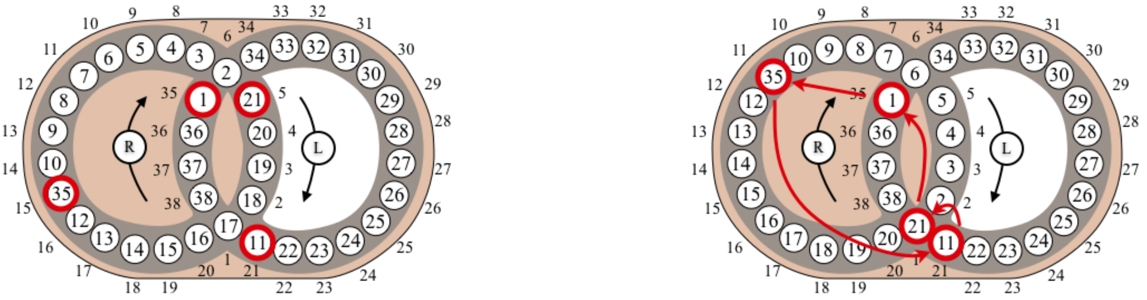
	prior to n^{th} move $RLR^{-1}L$:	**during n^{th} move $(RLR^{-1}L)^n$:**		**after n^{th} move $(RLR^{-1}L)^n$:**
n	**all disks that have moved off/on the left ring**	**disks that currently moved on and stayed on the left ring**	**disks that currently moved off/on the left ring**	**disks that are currently off the left ring**
1	$\emptyset$	35 (35) 21 (21)	6 (34) 1 (38)	7 (35) 2 (21)
2	$1, 6$	7 (35) 2 (21)	8 (34) 3 (38)	9 (35) 4 (21)
3	$1, 2, 3, 6, 7, 8$	9 (35) 4 (21)	10 (34) 5 (38)	11 (35) 35 (21)
2	$1, 2, 3, 4, 5, 6, 7, 8, 9, 10$			

Table 16.1: Summary of disks that moved off then back onto the left ring, and the positions affected, with first 3 application of $RLR^{-1}L$. The number in brackets next to the disk number represents the position the disk visited in the right ring.

procedure $RLR^{-1}L$ three more times to move the other 10 disks off the left ring. The result is shown in Figure 16.7a. Disks 9 through 10 were far enough away that they weren't affect, but disk 1 got sent to position 35. Notice most disks on the left ring are back in their proper order, so rotating them back to their home position by L^4 results in the 4-cycle:

$$\sigma_4 = (RLR^{-1}L)^3 L^5 (RLR^{-1}L)^3 L^4 = (1, 35, 11, 21).$$

We have just shown how to get a 4-cycle (see Figure 16.7b) and Theorem 16.2 tells us that this is probably the best we could do.



(a) $(RLR^{-1}L)^2L^5(RLR^{-1}L)^3$

(b) The fundamental 4-cycle: $\sigma_4 = (RLR^{-1}L)^3L^5(RLR^{-1}L)^3L^4 = (1, 35, 11, 21)$

Figure 16.7: The set-up moves for creating the 4-cycle $(1, 35, 11, 21)$, continued.

16.2.3 3-cycles

The fundamental 3-cycle, which we call σ_3, was built using compound commutators:

$$\sigma_3 = [[L^5, R^5], R^{-1}LR] = (6, 11, 12).$$

See Lecture 14 for the discussion.

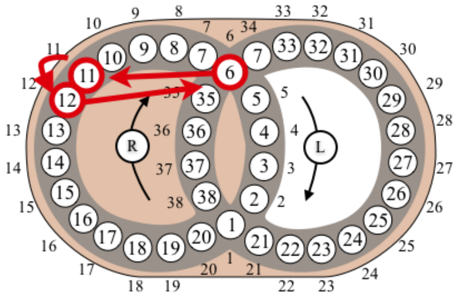


Figure 16.8: Fundamental 3-cycle: $\sigma_3 = [[L^5, R^5], R^{-1}LR] = (6, 11, 12)$.

16.2.4 2-cycles

Theorem 16.2 tells us that producing a 2-cycle is just as challenging as producing a 4-cycle since they are both odd. Luckily we already found a way to construct a 4-cycle:

$$\sigma_4 = (RLR^{-1}L)^3L^5(RLR^{-1}L)^3L^4 = (1, 35, 11, 21),$$

as shown in Figure 16.7b.

Using σ_4 we can construct the 2-cycle $(1, 11)$. Begin by applying σ_4. Now, if we can find a move sequence to swap disks 1 and 35, and swap disks 11 and 21 then we can produce the 2-cycle $(1, 11)$. To do this we can conjugate the pair of transpositions:

$$[L^5, R^5] = (1, 25)(6, 11)$$

by the four-step move sequence $\beta = RL^{-1}R^{-6}L$ which moves disks 1, 35, 11 and 21 to spots 6, 11, 1 and 25, respectively. Therefore,

$$\begin{aligned}\sigma_2 &= \sigma_4\beta[L^5, R^5]\beta^{-1}\\ &= ((RLR^{-1}L)^3L^5(RLR^{-1}L)^3L^4)(RL^{-1}R^{-6}L)(L^5R^5L^{-5}R^{-5})(L^{-1}R^6LR^{-1})\\ &= (1, 11).\end{aligned}$$

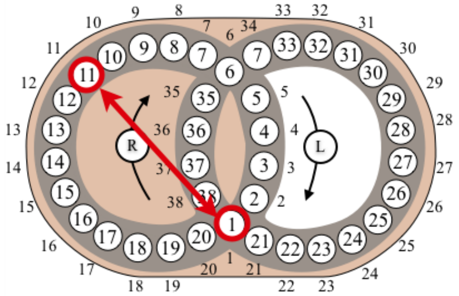


Figure 16.9: Fundamental 2-cycle: $\sigma_2 = (1, 11)$.

16.3 Solving the end-game

Theoretically, knowing how to perform 2-cycles is enough to solve the puzzle for any configuration. However, this would be very slow to perform manually. We now summarize a strategy for solving the puzzle.

(a) Starting with a scrambled puzzle put disks 7 through 16 on the left ring, and disks 21 through 30 on the right ring in proper numerical order.

(b) Still using general heuristics get a few more disks in their proper places if possible.

(c) Write down the remaining permutation in cycle form.

(d) Work on cycles that at length 5 or longer using conjugates of the fundamental 5-cycle $\sigma_5 = (1, 11, 35, 11, 16)$. If the cycle length is more than 5, you will be able to get 4 disks at a time into their right places. If the cycle is length 5 then you can solve all disks in the cycle this way.

(e) At this point all remaining cycles will be of length 5 or less. Using conjugates of the fundamental cycles: σ_5, σ_4, σ_3, σ_2 solve all disks in each cycle one cycle at a time.

16.4 Hungarian Rings - Coloured version

We now present a simple, and elegant strategy for solving the colour version of the puzzle shown in Figure 16.10.

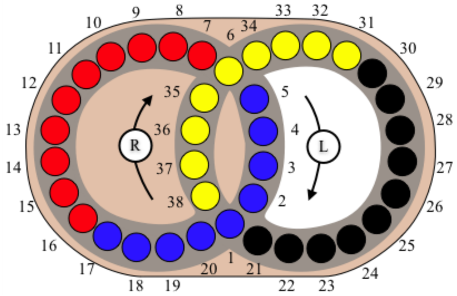


Figure 16.10: Hungarian Rings puzzle - coloured version.

There are 10 black disks and 10 red disks, but there are only 9 of each in blue and yellow. Solve the black and red disks first. There is enough room in the puzzle to do this using general heuristics. Once these are in their proper locations try to put as many blue and yellow disks in their home locations using general heuristics. To place the final remaining pieces (blue and yellow) you can swap two at a time, but if you're sneaky about how you do this then you actually don't need to use a 2-cycle. By placing the same coloured disks in spots 1 and 6, and the disks you want to swap in spots 11 and 30 (or 16 and 25), the commutators in Figure 16.3 can be used to swap pairs of disks. Since the intersection disks have the same colour this will go unnoticed, and this process will essentially allow you to swap any two blue and yellow disks. When using this method try to put either red or black disks in the intersection spots 1 and 6.

16.5 Exercises

1. Play with one of the virtual puzzles from the course website. The Hungarian Rings iphone app is pretty good. Try to solve each scrambling using the techniques developed in this section.

2. Show that for any cycle $\alpha = (a_1, a_2, a_3, a_4, a_5, \ldots, a_k)$ of length $k > 5$, there is a 5-cycle β so that $\alpha\beta$ has length $k - 4$. (This fact was used in the strategy for solving the end-game.)

Lecture 17

Partitions & Equivalence Relations

The cubies of Rubik's cube come in three types: corner cubies, edge cubies, and center cubies. In some sense we can think of any two edge cubies as equivalent since, using cube moves, we can take any edge cubie to the location of any other edge cubie (at the cost of possibly moving other pieces around). Similarly any two corner cubies are equivalent. Grouping similar elements together when trying to understand a large complicated set is a very powerful idea.

In this lecture we recall the concept of a *partition* of a set, and discuss its connection with the concept of an *equivalence relation* on a set.

17.1 Partitions of a Set

Consider the set of integers $\mathbb{Z}$. There are two well known subsets: the set of odd integers and the set of even integers. Every integer is a member of one of these subsets, and no integer is a member of both, so this gives a *partition* of $\mathbb{Z}$:

$$\mathbb{Z} = \{\dots - 5, -3, -1, 1, 3, 5, \dots\} \cup \{\dots - 4, -2, 0, 2, 4, \dots\}.$$

Definition 17.1: A **partition** of a set A is a finite collection of non-empty subsets $A_1, A_2, \dots, A_n$ satisfying the following properties.

(a) A is the union of all the A_i's: $A = A_1 \cup A_2 \cup \cdots \cup A_n$,

(b) the A_i's are disjoint: $A_i \cap A_j = \emptyset$ for all $i \neq j$, $1 \leq i, j \leq n$.

Example 17.1: Let E be the set of edge cubies of Rubik's cube, let V be the set of corner cubies, and let C be the set of centre cubies. E, V and C are disjoint sets, and their union is the set of all cubies. Therefore $E \cup V \cup C$ is a partition of the set of all cubies.

Example 17.2: (a) The three sets

$$\begin{aligned} A_0 &= \{\ldots-9,-6,-3,0,3,6,9,\ldots\} = \{3k \mid k\in\mathbb{Z}\},\\ A_1 &= \{\ldots-8,-5,-2,1,4,7,10\ldots\} = \{3k+1 \mid k\in\mathbb{Z}\},\\ A_2 &= \{\ldots-7,-4,-1,2,5,8,11\ldots\} = \{3k+2 \mid k\in\mathbb{Z}\},\end{aligned}$$

form a partition of the integers $\mathbb{Z}$. A_0 is all the integers which are divisible by 3, A_1 are those integers whose remainder is 1 when divided by 3, and A_2 are those whose remainder is 2 when divided by 3. These exhaust all the possibilities of the remainder, and so $A_0 \cup A_1 \cup A_3 = \mathbb{Z}$. Moreover, for any particular integer, the remainder (upon division by 3) is unique so these sets are disjoint.

(b) A partition of the positive integers $\mathbb{Z}^+$ into two sets is $P \cup \overline{P}$ where P is the set of prime numbers, and $\overline{P} = \mathbb{Z}^+ - P$ is the set of non-prime positive integers.

(c) The sets $\{0,1,2\}$ and $\{2,3,4\}$ do not form a partition of $\mathbb{Z}_5 = \{0,1,2,3,4\}$ since they are not disjoint. They have the element 2 in common.

We partitioned $\mathbb{Z}$ in three different ways: (i) into odd and even sets, (ii) into sets where the remainder upon division by 3 were the same, and (iii) into the set of primes, and non-primes. This illustrates there is more than one way to partition a set. As for which one to use, this depends on the problem you are trying to solve.

Partitioning a set gives us a nice way to group together elements with similarities. This allows us to focus our attention on subsets rather than the whole set, and this comes in handy when dealing with permutation puzzles. Partitions are closely related to another concept known as an *equivalence relation*. We now introduce this concept and show its connection with partitions.

17.2 Relations

We are familiar with many types of relations: "parent", "brother", "sister", "sibling", "spouse", $<$, $=$, $>$, $\subset$, and other types of comparisons. In essence what we are doing is comparing two objects from the same set.

Definition 17.2: Let A be a set. A subset $\mathcal{R} \subset A \times A$ is called a **relation on** A. If $(x,y) \in \mathcal{R}$ then we say x is related to y (and we sometimes write $x\mathcal{R}y$ for simplicity).

Notice this definition is quite basic. It just says that by a "relation" we just mean a subset of $A \times A$. Any such subset will be a relation.

Example 17.3: Let $A = \{1,2,3,4,5\}$, then each of the following is a relation on A.

(a) $\mathcal{R}_1 = \{(1,4),(3,2)\}$

(b) $\mathcal{R}_2 = \{(1,1),(2,2),(3,3),(4,4),(5,5)\} = \{(a,b) \in A \times A \mid a = b\}$

(c) $\mathcal{R}_3 = \{(1,2),(1,3),(1,4),(1,5),(2,3),(2,4),(2,5),(3,4),(3,5),(4,5)\} = \{(x,y) \in A \times A \mid x < y\}$

In relation $\mathcal{R}_1$ we say: 1 is related to 4 and 3 is related to 2. But 1 is not related to 2. Also, 4 is not related to 1 in this case since $(4,1) \notin \mathcal{R}_1$. Read this carefully, 1 IS related to 4, but 4 IS NOT related to 1. Order matters in a relation. For example, John is the father of Jack, but Jack is not the father of John. This subtlety won't bother us too much (we are more interested in equivalence relations, which are symmetric, as discussed in the next section).

Since, by definition, a relation is a subset of $A \times A$, and $|A \times A| = 5^2 = 25$ then there are 2^{25} possible relations on A (each element of $|A \times A|$ can either be included in the relation, or not, hence there are two choices for each element). Some relations, of course, are more interesting than others.

Example 17.4: Let $A = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$ (that is, A is the set of all subset of $\{1,2\}$). Consider the relation

$$\mathcal{R} = \{(\emptyset,\ \emptyset),\ (\emptyset, \{1\}),\ (\emptyset, \{2\}),\ (\emptyset, \{1,2\}),\ (\{1\}, \{1\}),\ (\{1\}, \{1,2\}),\ (\{2\}, \{2\}),\ (\{2\}, \{1,2\}),\ (\{1,2\}, \{1,2\})\}.$$

This is an example of the "subset" relation, since $(X, Y) \in \mathcal{R}$ precisely when $X \subset Y$.

Example 17.5: Let $\mathcal{C}$ be the set of all the different configurations of Rubik's cube (that is, all the ways to mess up a cube). Let's say two configurations X and Y are related if there is a quarter turn of one of the 6 faces which takes configuration X to configuration Y:

$$(X, Y) \in \mathcal{R} \quad \text{if} \quad Y \text{ can be obtained from } X \text{ by a quarter turn of one face.}$$

This defines a relation on $\mathcal{C}$. The cubes in Figures 17.1a and 17.1b are related (by a quarter turn of the r face), and the cubes in 17.1b and 17.1c are related (by a quarter turn of the u face). However, the cubes in 17.1a and 17.1c are not related, since it takes two face turns to get from one configuration to the other.

Note that if $(X, Y) \in \mathcal{R}$ then $(Y, X) \in \mathcal{R}$, since each quarter turn has an inverse. In this case we would say $\mathcal{R}$ is a *symmetric* relation.

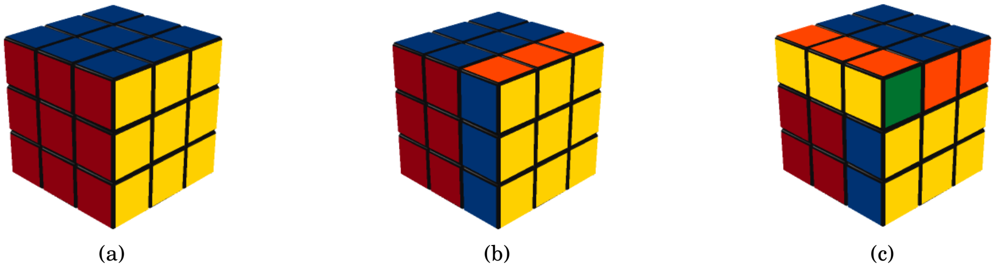


Figure 17.1: Three different configurations of Rubik's cube.

17.3 Equivalence Relation

For a given set, some relations are more useful than others. We saw in Example 17.3 that there are 2^{25} different possible ways to define a relation on $A = \{1, 2, 3, 4, 5\}$, but relations (b) and (c) seem much more useful (or should we say meaningful) than relation (a). Perhaps this is because we are so familiar with the relations "$=$" and "$<$". In this section we focus our attention on a special type of relation that is very useful in mathematics.

First a digression into relationships amongst people. For this let's just consider the set of all people who are currently alive, call this set $\mathcal{P}$. There are a number of relations we can consider on $\mathcal{P}$, for example if we are interested in who is whose child then the relation we would consider is: *$x\mathcal{R}y$ if x is a child of y*. Or maybe we want to consider the relationship of being a brother: *$x\mathcal{R}y$ if x is a brother of y*. Perhaps we just want to know who is married, and to whom: *$x\mathcal{R}y$ if x is a spouse of y*. If your interest is in relationships on a more global scale then you can consider a proximity relation: *$x\mathcal{R}y$ if x lives in the same city as y*.

There are some differences in the behaviour of these relations. Consider the "brother of" relation. *Tim could be a brother of Alice*, but (assuming Alice is female) *Alice is not a brother of Tim*. We say that $\mathcal{R}$ is not *symmetric* in this case. However, the "spouse of" relation is *symmetric*: *if X is the spouse of Y then Y is the spouse of X*.

For the "proximity" relation, *if X lives in the same city as Y* and *Y lives in the same city as Z*, then it should follow that *X lives in the same city as Z*. We refer to this property as *transitivity*. Notice the "child relation" is not transitive, since *if Emma is a child of Karen*, and *Karen is a child of Henry*, then *Emma is not a child of Henry* (at least we hope not).

Another property that some relations may possess is the ability for an element to be related to itself. For example, *X lives in the same city as X* is certainly true. But, *X is a child of X* is impossible (though this would make a disturbing plot for some science fiction movie). A relation where all elements are related to themselves is known as *reflexive*.

An important, and very useful, class of relations are the relations that are *reflexive*, *symmetric* and *transitive*.

Definition 17.3: Let $\mathcal{R}$ be a relation on a set A. We call $\mathcal{R}$ an **equivalence relation** on A if it satisfies the following properties:

(a) Each element is related to itself: $(a, a) \in \mathcal{R}$ for all $a \in A$ (reflexive property)

(b) If a is related to b then b is related to a: $(a, b) \in \mathcal{R}$ implies $(b, a) \in \mathcal{R}$ (symmetric property)

(c) If a is related to b, and b is related to c then a is related to c: $(a, b) \in \mathcal{R}$ and $(b, c) \in \mathcal{R}$ implies $(a, c) \in \mathcal{R}$ (transitive property).

Notation: If $\mathcal{R}$ is an equivalence relation on A then we often write $x \equiv y$, or $x \sim y$ in place of $(x, y) \in \mathcal{R}$ for simplicity.

The "child of", "brother of", and "spouse of" relations are not equivalence relations. To see why we just need to observe that one of the three properties doesn't hold. In each case the reflexive property fails to hold. However, the "proximity" relation is an equivalence relation.

In Example 17.3 the relations $\mathcal{R}_1$ and $\mathcal{R}_3$ are not equivalence relations. For instance, neither one is symmetric. However, $\mathcal{R}_2$ is an equivalence relation.

The "proximity" relation $\sim$ on $\mathcal{P}$ is an equivalence relation. Pick some person, say person X from *Vancouver*. What does the set of all people related to X represent: $\{Y \in \mathcal{P} \mid Y \sim X\}$? Well, this would consist of all the people who live in Vancouver. Think about why? Sets of this type will be important for us, so we give them a special name.

Definition 17.4: Let $\sim$ be an equivalence relation on a set A. For each $a \in A$ the set

$$[a] = \{x \in A \mid x \sim a\}$$

is called the **equivalence class of A containing** a. We call a a **representative** of the equivalence class $[a]$. [a]

[a] The equivalence class of a is sometimes denoted by $[a]_{\mathcal{R}}$ or $[a]_{\sim}$.

Lemma 17.1: *If $\sim$ is an equivalence relation on a set A and $x, y \in A$, then*

(a) $x \in [x]$ (an equivalence class contains its representative)

(b) $x \sim y$ if and only if $[x] = [y]$ (if two elements are related then their equivalence classes are equal)

(c) $[x] = [y]$ or $[x] \cap [y] = \emptyset$ (equivalence classes are either equal or disjoint).

Proof: (a) Since $\sim$ is reflexive $x \sim x$, therefore $x \in [x]$.

(b) Suppose $x \sim y$. We want to show that this implies $[x] = [y]$. To do this, let $z \in [x]$, then $z \sim x$ and since $x \sim y$ it follows that $x \sim y$, by the transitive property, and so $z \in [y]$. Therefore $[x] \subset [y]$. Moreover, $y \sim x$ by symmetry and a similar argument shows $[y] \subset [x]$. Therefore $[x] = [y]$.

Conversely, suppose $[x] = [y]$. By part (a), $x \in [x] = [y]$, and so $x \sim y$.

(c) If $[x] \cap [y] \neq \emptyset$ then let $z \in [x] \cap [y]$. It follows that $z \sim x$ and $z \sim y$, and so $x \sim y$ by transitivity. Now applying part (b) we have $[x] = [y]$. □

Partitions and equivalence relations are related as the next result suggests.

Theorem 17.2: *(a) If A is a set and $\mathcal{R}$ is an equivalence relation on A then the set of equivalence classes form a partition of A.*

(b) If $A_1, \ldots, A_n$ is a partition of a set A then the relation $\mathcal{R}$ defined by

$$a\mathcal{R}b \quad \text{if} \quad a, b \in A_i \text{ for some } i,$$

is an equivalence relation on A. This relation can written as

$$\mathcal{R} = \bigcup_{i=1}^{n} A_i \times A_i.$$

The sets A_i are the equivalence classes of relation $\mathcal{R}$.

Proof: (a) This is a direct consequence of Lemma 17.1.

(b) By definition of $\mathcal{R} = \bigcup_{i=1}^{n} A_i \times A_i$ symmetric. Reflexivity follows from the fact that A is the union of the A_i's, and transitivity follows from the fact that the A_i's are disjoint. □

Definition 17.5: If $\sim$ is an equivalence relation on a set A, then a **set of class representatives** is a subset of A which contains exactly one element from each equivalence class. We denote the set of class representative by $A/\sim$.

If $\sim$ is an equivalence relation on a set A , and $x \sim y$ then we say x and y are **equivalent**, rather than simply saying they are related.

Let's look at some examples to get a little more comfortable with these ideas.

Example 17.6 (Congruence relation on $\mathbb{Z}$): Let n be a positive integer. Define an equivalence relation $\equiv$ on $\mathbb{Z}$ by

$$a \equiv b \quad \text{if } a - b \text{ is divisible by } n.$$

We say a is **congruent to b modulo n** and write $a \equiv b \pmod{n}$. In Exercise 2 you are asked to verify that this is indeed and equivalence relation.

For example, $26 \equiv 4 \pmod{11}$ since $26 - 4 = 22$ is divisible by 11. We say 26 is equivalent to 4 modulo 11. On the other hand, $7 \not\equiv 3 \pmod{5}$ since 5 does not divide $7 - 3 = 4$.

The equivalence class of x modulo n is often called the **congruence class of x modulo n**.

The equivalence relation $\equiv \pmod{2}$ on $\mathbb{Z}$ has two equivalence (congruence) classes:

$$[0] = \{0, \pm 2, \pm 4, \ldots\} \quad \text{and} \quad [1] = \{\pm 1, \pm 3, \pm 5, \ldots\}$$

A set of equivalence class representatives is $\{0, 1\}$.

The equivalence relation $\equiv \pmod{3}$ on $\mathbb{Z}$ has three equivalence (congruence) classes:

$$[0] = \{0, \pm 3, \pm 6, \ldots\}\,, \quad [1] = \{\pm 1, \pm 4, \pm 7, \ldots\} \quad \text{and} \quad [2] = \{\pm 2, \pm 5, \pm 8, \ldots\}$$

A set of equivalence class representatives is $\{0, 1, 2\}$.

In general, for $n \in \mathbb{Z}^+$ and $a \in \mathbb{Z}$, the class of a is

$$[a] = \{a + kn \mid k \in \mathbb{Z}\}.$$

The set of equivalence class representatives (also called congruence class representatives modulo n) is

$$(\mathbb{Z}/\equiv) = \{0, 1, 2, \ldots, n-1\}.$$

Example 17.7: Let $\mathcal{C}$ be the set of all the different configurations of Rubik's cube. The relation on $\mathcal{C}$ given in Example 17.5 is not transitive as we saw in that example.

Instead, let's consider another relation on $\mathcal{C}$ defined by $X \equiv Y$ if there is a sequence of moves involving only U and R that takes configuration X to configuration Y. This is an equivalence relation. Check for yourself that the three properties hold.

The 3 configurations shown in Figure 17.1 are equivalent, and therefore are elements of the same equivalence class. A representative for this class is the solved cube 17.1a. How many other configurations are equivalent to the solved cube? It turns out that there are a whopping $73,483,200$ configurations all equivalent to the solved cube. This means that by only twisting the R and U faces of the cube, you can generate over 73 million different configurations of the cube.

Sage

```
 sage: S48=SymmetricGroup(48)
 sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
 sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
 sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
 sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
 sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
 sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
 sage: H=S48.subgroup([R,U])
 sage: H.order()
73483200
```

Example 17.8: Let $\mathcal{A}$ denote the set of all possible ways to reassemble Rubik's cube. That is, first you take it apart, then put it back together in the shape of a cube again. Define a relation $\sim$ on $\mathcal{A}$ as follows:

$$X \sim Y \quad \text{if} \quad \text{through a sequence of legal cube moves (i.e. twists of the } 6 \text{ faces), } X \text{ can be taken to } Y.$$

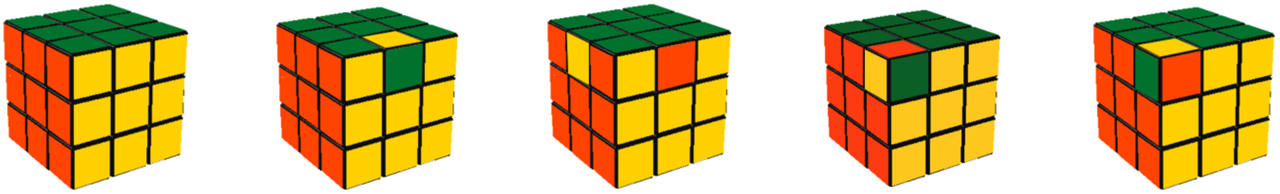
All that this means is we consider two cubes equivalent if one can be twisted into the other.

What is the equivalence class of the solved cube?

This is really asking, what configurations are equivalent to the solved state configuration? In other words, what are all the possible configurations one can achieve from the solved cube by twisting faces. In this context, where we are considering all assembled cubes $\mathcal{A}$, this is an interesting question, since if the equivalence class is not all of $\mathcal{A}$ it means there are ways to reassemble the cube which are not solvable. So you can mess with your friends cube by taking it apart and reassembling it into an unsolvable cube.

Using the notation introduced in this section, and letting X_0 denote the cube in the solved state, then what we want to know is $[X_0]$. Moreover, if there is more than one equivalence class then it would be interesting to know how many there are and a set of equivalence class representative, i.e. $\mathcal{A}/\sim$.

We will investigate this question in Lecture 20. But for now we'll note that $|\mathcal{A}/\sim| \geq 5$ since Figure 17.2 shows five assemblies of Rubik's cube which are not equivalent under legal cube moves.



(a) solved

(b) edge flip

(c) edge swap

(d) clockwise corner twist

(e) counterclockwise corner twist

Figure 17.2: Five different equivalence class representatives of $\mathcal{A}$. How many more are there?

We also know that a corner swap, see Figure 17.3, is not equivalent to X_0.

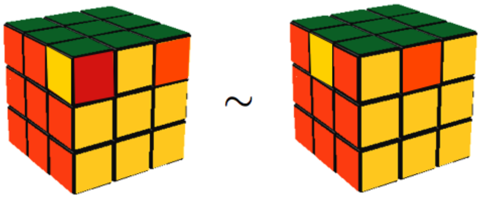


Figure 17.3: A corner swap is equivalent to an edge swap, but not equivalent to the solved state.

However, it is equivalent to the "edge swap" in Figure 20.6b. We'll see this when we study the Fundamental Theorem of Cubology in Lecture 20.

17.4 Exercises

1. Consider the "cousin of" relation:

$$x\mathcal{R}y \quad \text{if } x \text{ is a cousin of } y.$$

Is $\mathcal{R}$ symmetric? Is it transitive?

2. In Example 17.6 it was stated that $\equiv \pmod{n}$ is an equivalence relation on $\mathbb{Z}$. Prove this statement. That is, show it is reflexive, symmetric and transitive.

3. For each the following relations defined on the set X determine whether or not the relation is reflexive, symmetric, or transitive.

   (a) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $a \mid b$ (i.e. a divides b)

   (b) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $a + b = 10$

   (c) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $a - b > 0$

   (d) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $a + b$ is even

   (e) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $a - b$ is even

   (f) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $3 \mid a + b$

   (g) $X = \mathbb{Z}$, $a\mathcal{R}b$ if $\gcd(a, b) = 1$

   (h) $X = \mathbb{Z} \times (\mathbb{Z} - \{0\})$, $(a, b)\mathcal{R}(c, d)$ if $ad = bc$

   (i) $X = \mathbb{R} \times \mathbb{R}$, $(a, b)\mathcal{R}(c, d)$ if $\sqrt{(a-c)^2 + (b-d)^2} \leq 1$

   (j) $X = \mathbb{R} \times \mathbb{R}$, $(a, b)\mathcal{R}(c, d)$ if $ac + bd = 0$

4. Define the relation $\mathcal{R}$ on $\mathbb{R} \times \mathbb{R}$ by

   $$(a, b)\mathcal{R}(c, d) \quad \text{if} \quad b - a = d - c.$$

   Show that $\mathcal{R}$ is an equivalence relation and describe the set $\mathcal{R}$ geometrically.

5. Define the relation $\mathcal{R}$ on $\mathbb{R} \times \mathbb{R}$ by

   $$(a, b)\mathcal{R}(c, d) \quad \text{if} \quad a^2 + b^2 = c^2 + d^2.$$

   Show that $\mathcal{R}$ is an equivalence relation and describe the set $\mathcal{R}$ geometrically.

6. Define the relation $\mathcal{R}$ on $X = \{1, 2, 3, \ldots, 20\}$ by

   $$a\mathcal{R}b \quad \text{if} \quad 3 \mid a - b.$$

   Show that $\mathcal{R}$ is an equivalence relation. Describe the equivalence classes of the corresponding partition of X.

7. Define the relation $\mathcal{R}$ on $X = \{1, 2, 3, \ldots, 20\}$ by

   $$a\mathcal{R}b \quad \text{if} \quad a \text{ and } b \text{ have the same prime divisors.}$$

   Show that $\mathcal{R}$ is an equivalence relation. Describe the equivalence classes of the corresponding partition of X.

8. For each of the following statements about relations on a set A, where $|A| = n$, determine whether the statement is true or false. If it is false, give a counterexample.

   (a) If $\mathcal{R}$ is a reflexive relation on A, then $|\mathcal{R}| \geq n$.

   (b) If $\mathcal{R}$ is a relation on A and $|\mathcal{R}| \geq n$, then $\mathcal{R}$ is reflexive.

   (c) If $\mathcal{R}_1$, $\mathcal{R}_2$ are relations on A and $\mathcal{R}_1 \subset \mathcal{R}_2$, then $\mathcal{R}_1$ reflexive (symmetric, transitive) $\Rightarrow$ $\mathcal{R}_2$ reflexive (symmetric, transitive).

   (d) If $\mathcal{R}_1, \mathcal{R}_2$ are relations on A and $\mathcal{R}_1 \subset \mathcal{R}_2$, then $\mathcal{R}_2$ reflexive (symmetric, transitive) $\Rightarrow$ $\mathcal{R}_1$ reflexive (symmetric, transitive).

   (e) If $\mathcal{R}$ is and equivalence relation on A, then $n \leq |\mathcal{R}| \leq n^2$.

9. If $A = \{a, b, c, d\}$, determine the number of relations on A that are (i) reflexive, (ii) symmetric, (iii) reflexive and symmetric, (iv) reflexive and contains (a, b), (v) symmetric and contains (a, b).

10. If $A = \{1, 2, 3, 4\}$, give and example of a relation $\mathcal{R}$ on A that is
    (a) reflexive and symmetric, but not transitive.
    (b) reflexive and transitive, but not symmetric.
    (c) symmetric and transitive, but not reflexive

11. Describe a partition of the set of all prime numbers into four classes.

12. What is wrong with the following argument?

    Let A be a set and $\mathcal{R}$ a relation on A. If $\mathcal{R}$ is symmetric and transitive, then $\mathcal{R}$ is reflexive.

    Proof: Let $(x, y) \in \mathcal{R}$. By the symmetric property $(y, x) \in \mathcal{R}$. Then with $(x, y), (y, x) \in \mathcal{R}$, it follows by the transitive property that $(x, x) \in \mathcal{R}$. Consequently $\mathcal{R}$ is reflexive. □

13. Let A be a set with $|A| = n$, and let $\mathcal{R}$ be an equivalence relation on A with $|\mathcal{R}| = r$. Why is $r - n$ always even?

14. **Conjugation is an equivalence relation.** Let G be a group, show that the relation

    $$g\mathcal{R}h \iff g \text{ is a conjugate of } h,$$

    is an equivalence relation.

15. Let G be a group and H a subgroup of G. Define a relation $\mathcal{R}$ on G by

    $$a\mathcal{R}b \quad \text{if} \quad b^{-1}a \in H.$$

    (a) Show $\mathcal{R}$ is an equivalence relation.
    (b) Show that each equivalence class $[a]$ has the form $aH = \{ah \mid h \in H\}$ for some a. The is called the *left coset of H in G containing a.*
    (c) Show that each equivalence class has the same cardinality. That is, show $|aH| = |bH|$, for any $a, b \in H$.
    (d) Conclude from Theorem 17.2 that $|H|$ divides $|G|$. This proves *Lagrange's Theorem*: the order of a subgroup divides the order of a group. (We'll discuss Lagrange's Theorem further in Section 18.2.)

16. Consider the set of all 2×2 matrices with real entries:

    $$M_{2,2}(\mathbb{R}) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mid a, b, c, d \in \mathbb{R} \right\}.$$

    Define a relation $\mathcal{R}$ on $M_{2,2}(\mathbb{R})$ by

    $$A\mathcal{R}B \quad \text{if} \quad A \text{ is row equivalent to } B.$$

    (By *row equivalent* we mean A can be converted to B through elementary row operations: (i) multiply a row by a scalar, (ii) swap two rows, (iii) add a multiple of another row to an existing row.)
    Show $\mathcal{R}$ is an equivalence relation. How many equivalence classes are there? Determine a set of class representatives.

17. Define a relation $\mathcal{R}$ on $M_{2,2}(\mathbb{R})$ by

    $$A\mathcal{R}B \quad \text{if} \quad \text{there exists and invertible matrix } C \text{ such that } B = CA.$$

    Show $\mathcal{R}$ is an equivalence relation. How does this relation compare to the one in Exercise 16.

Lecture 18

Cosets & Lagrange's Theorem

In this lecture we introduce a powerful tool for analyzing a group - a *coset*. We'll then use cosets to prove Lagrange's Theorem which states the size of a subgroup divides the size of the group.

18.1 Cosets

Let H be a subgroup of a group G. Define a relation $\sim_H$ on G as follows:

$$a \sim_H b \iff a^{-1}b \in H. \tag{18.1}$$

Equivalently, $a \sim_H b$ if and only if $a^{-1}b = h$ for some $h \in H$. Or another way to say this is $a \sim_H b$ if and only if $b = ah$ for some $h \in H$.

Lemma 18.1: *If $H < G$, then $\sim_H$ is an equivalence relation on G. Moreover, if $[a]$ denotes the equivalence class of $a \in G$, then*

$$[a] = \{ah \mid h \in H\}.$$

Proof: We need to show $\sim_H$ is reflexive, symmetric and transitive. For all $a, b, c \in G$:

Reflexive: Since H is a subgroup it contains the identity, so $a^{-1}a = e \in H$, Therefore, $a \sim_H a$.

Symmetric: If $a \sim_H b$ then $a^{-1}b \in H$. Since H is a subgroup it is closed under taking inverses, so $(a^{-1}b)^{-1} = b^{-1}a \in H$. Therefore $b \sim_H a$.

Transitive: If $a \sim_H b$ and $b \sim_H c$ then $a^{-1}b, b^{-1}c \in H$. Since H is a subgroup it is closed under products, so $(a^{-1}b)(b^{-1}c) = a^{-1}c \in H$. Therefore $a \sim_H c$.

It follows that $\sim_H$ is an equivalence relation on G.

Since $a \sim_H b$ if and only if $b = ah$ for some $h \in H$, then

$$\begin{aligned}[a] &= \{b \mid a \sim_H b\} \\ &= \{ah \mid h \in H\}\end{aligned}$$

□

The following definition gives a name to the particular type of equivalence class that appeared in the lemma.

Definition 18.1 (Coset of H in G): Let G be a group and H a subgroup of G. For any $a \in G$, the set

$$aH = \{ah \mid h \in H\}$$

is called the **left coset of** H in G containing a. Analogously,

$$Ha = \{ha \mid h \in H\}$$

is called the **right coset of** H in G containing a. The element a is called the **coset representative of** aH or Ha.

The *right coset* is the equivalence class that comes from the equivalence relation $a \sim b$ if and only if $ab^{-1} \in H$.

Since left cosets of H are the equivalence classes under the relation $\sim_H$ they form a partition of the group G. In particular, for any two left cosets aH and bH we either have

$$aH = bH \quad \text{or} \quad aH \cap bH = \emptyset.$$

Let's see what these cosets look like in a few specific examples.

Example 18.1: Let $S_3 = \{\varepsilon, (1,2), (1,3), (2,3), (1,2,3), (1,3,2)\}$, and consider the subgroup $H = \langle(1,2)\rangle = \{\varepsilon, (1,2)\}$. The left cosets of H are:

$$\begin{aligned} \varepsilon H &= H = \{\varepsilon, (1,2)\} \\ (1,3)H &= \{(1,3), (1,3)(1,2)\} = \{(1,3), (1,3,2)\} \\ (2,3)H &= \{(2,3), (2,3)(1,2)\} = \{(2,3), (1,2,3)\} \end{aligned}$$

The left coset representatives of H in G are therefore ε, $(1,3)$, and $(2,3)$.

Notice that

$$(1,2)H = H, \quad (1,3,2)H = (1,3)H, \quad (1,2,3)H = (2,3)H.$$

In other words, it doesn't matter which element of the coset you use to describe it. For instance, $(1,2), (1,3,2), (1,2,3)$ is another set of left coset representatives of H in G.

The right cosets of H are:

$$\begin{aligned} H\varepsilon &= H = \{\varepsilon, (1,2)\} \\ H(1,3) &= \{(1,3), (1,2)(1,3)\} = \{(1,3), (1,2,3)\} \\ H(2,3) &= \{(2,3), (1,2)(2,3)\} = \{(2,3), (1,3,2)\} \end{aligned}$$

Notice that the left and right cosets are not necessarily the same. For example $(1,3)H \neq H(1,3)$.

For the subgroup $K = \langle(1,2,3)\rangle = \{\varepsilon, (1,2,3), (1,3,2)\}$ there are only two distinct left cosets:

$$\begin{aligned} K &= \{\varepsilon, (1,2,3), (1,3,2)\} \\ (1,2)K &= \{(1,2), (1,2)(1,2,3), (1,2)(1,3,2)\} = \{(1,2), (1,3), (2,3)\}. \end{aligned}$$

Notice that $K = (1,2,3)K = (1,3,2)K$ and $(1,2)K = (1,3)K = (2,3)K$.

Example 18.2: Consider $\mathbb{Z}_{12}$, the group of integers modulo 12, and the subgroup $H = \langle 3 \rangle = \{0, 3, 6, 9\}$. The cosets of H are:

$$\begin{aligned} 0 +_{12} H &= H = \{0,3,6,9\} \\ 1 +_{12} H &= \{1,4,7,10\} \\ 2 +_{12} H &= \{2,5,8,11\} \end{aligned}$$

Note that the left and right cosets are the same in this case since $\mathbb{Z}_{12}$ is abelian. Also,

$$1 +_{12} H = 4 +_{12} H = 7 +_{12} H = 10 +_{12} H.$$

In each of the examples above notice that the only coset of H which is a subgroup of G is H itself. Here are some basic properties of cosets.

Lemma 18.2 (Properties of Cosets): *Let H be a subgroup of G and $a \in G$.*

(a) $a \in aH$

(b) $aH = H \iff a \in H$

(c) For $a, b \in G$, either $aH = bH$ or $aH \cap bH = \emptyset$.

(d) $aH = bH \iff a^{-1}b \in H \iff b^{-1}a \in H$

(e) If H is finite then $|aH| = |H|$

(f) $aH = Ha \iff a^{-1}Ha = H.$

(Note that by $a^{-1}Ha$ we mean the set $\{a^{-1}ha \mid h \in H\}$.)

Proof: First observe that since aH is the equivalence class $[a]$ then (a), (c), and (d) are just the results of Lemma 17.1 which we have already proven.

(b) If $aH = H$ then $a \in aH = H$. Conversely, suppose $a \in H$. Then $aH \subset H$, while on the other hand, if $b \in H$ then $a^{-1}b \in H$ so $b \in aH$. Therefore $aH = H$.

Another way to prove this is to just observe that it is a special case of (d) where $b = e$. Therefore it follows as a direct consequence of Lemma 17.1.

(e) The map $\psi : H \to aH$ defined by

$$\psi(h) = ah,$$

is a bijection.
Injective: $\psi(h_1) = \psi(h_2)$ implies $ah_1 = ah_2$, and by cancellation, $h_1 = h_2$.
Surjective: For $b \in aH$, there is an $h \in H$ such that $b = ah$. Therefore, $a^{-1}b \in H$ and $\psi(a^{-1}b) = b$.

Since ψ is a bijection then H and aH must have the same size: $|H| = |aH|$.

(f) ($\Longrightarrow$) If $aH = Ha$ then for any $h \in H$ there is an $x \in H$ such that $ax = ha$, so $a^{-1}ha \in H$. Therefore $a^{-1}Ha \subset H$. On the other hand, for any $h \in H$ there is a $y \in H$ such that $ah = ya$, so $h = a^{-1}ya \in a^{-1}Ha$. Therefore $H \subset a^{-1}Ha$. It follows that $H = a^{-1}Ha$.

($\Longleftarrow$) If $a^{-1}Ha = H$ then for any $h \in H$ there is an $x \in H$ such that $a^{-1}xa = h$, so $ah = xa \in Ha$. Therefore $aH \subset Ha$. A similar argument shows $Ha \subset Ha$. Therefore $aH = Ha$. □

18.2 Lagrange's Theorem

We stated Lagrange's Theorem back in Lecture 11. Now we have the tools to prove it.

Theorem 18.3 (Lagrange's Theorem): *If G is a finite group and H is a subgroup of G, then $|H|$ divides $|G|$.*

Proof: Let $\sim_H$ be the equivalence relation on G defined in (18.1). Then the equivalence classes are the left cosets $[a] = aH$. Let

$$a_1H, a_2H, \ldots, a_kH$$

denote the distinct left cosets of H in G. By Lemma 18.2(e) all equivalence classes have the same size: $|[a_i]| = |a_iH| = |H|$. Since these classes partition G then

$$G = a_1H \cup a_2H \cup \cdots \cup a_kH, \qquad \text{(disjoint union)}$$

and so

$$|G| = |a_1H| + |a_2H| + \cdots + |a_kH| = k|H|, \tag{18.2}$$

(by Theorem 2.1). Therefore $|H|$ divides $|G|$. □

From Equation (18.2) we have a formula for the number of left cosets of H in G:

$$\text{number of left cosets} \quad = \quad \text{number of } \sim_H \text{ equivalence classes} \quad = \quad \frac{|G|}{|H|}.$$

Similarly, working with right cosets rather than left cosets in our previous arguments, we have that the number of right cosets is also $|G|/|H|$.

In particular, the *number* of left and right cosets of a given subgroup are the same. This is an important number in calculations involving groups and is called the **index of H in G**, which is denoted by $[G/H]$:

$$[G/H] := \text{ the index of } H \text{ in } G \; = \frac{|G|}{|H|}. \tag{18.3}$$

However, even though the *number* of left and right cosets of a subgroup H in G is the same, the actual left and right cosets themselves can be different. See Example 18.1.

In Lecture 11 we noted a few consequences of Lagrange's Theorem. We'll list them here again for convenience.

Corollary 18.4 ($ord(a)$ divides $|G|$): *Let G be a finite group and $a \in G$. Then*

(a) $ord(a)$ *divides* $|G|$.

(b) $a^{|G|} = e$.

Example 18.3 (Number of different cubes up to U, R moves): In Example 17.7 we considered the set $\mathcal{C}$ of all the different configurations of Rubik's cube and the equivalence relation $\equiv$ on $\mathcal{C}$ defined by

$$X \equiv Y \iff \begin{array}{l}\text{if there is a sequence of moves involving only } U \text{ and } R \\ \text{that takes configuration } X \text{ to configuration } Y.\end{array}$$

If we identify each configuration in $\mathcal{C}$ with its corresponding permutation in RC_3, then the equivalence relation $\equiv$ can be described as

$$X \equiv Y \iff X^{-1}Y \in H = \langle U, R\rangle$$

In other words, it is just the relation $\sim_H$, and so the equivalence classes are the cosets of $H = \langle U, R\rangle$.

If X_0 denotes the cube in the solved state, then $[X_0] = H$, and as we found in Example 17.1, has size $73,483,200$. The number of distinct equivalence classes is given by (18.3), and we can use Sage to compute it.

Sage

```
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
sage: RC3=S48.subgroup([R,L,U,D,F,B])
sage: H=S48.subgroup([R,U])
sage: RC3.order()/H.order()
588597166080
```

What does this mean? It means that if we think of any two configurations, in which one can be obtained from the other by twisting only the R and U faces, as equivalent, then we've partitioned $\mathcal{C}$ into $588,597,166,080$ sets, each of size $73,483,200$, where within each set of the partition any two configurations are equivalent under U, R moves. But for any two configurations coming from different sets in the partition, there is no way to obtain one from the other using U, R moves. In this sense there are $588,597,166,080$ *different* cubes up to R, U moves.

An Application to Number Theory:

We briefly look at how our previous results can be used to establish two very famous theorems in number theory.

Corollary 18.5 (Fermat's Little Theorem): *For every integer a and every prime p,*

$$a^p \equiv a \pmod{p}.$$

That is, p divides $a^p - a$.

Proof: Let r be the remainder of a upon division by p. Since $a \equiv r \pmod{p}$ and $a^p \equiv r^p \pmod{p}$ then it suffices to prove the corollary for $0 \leq a \leq p-1$. The result for $a = 0$ is trivial. So assume $1 \leq a \leq p-1$. Then we can assume $a \in U(p)$, the group of integers $\{1, 2, \ldots, p-1\}$ under multiplication modulo p. (See Lecture 10 for further discussion of $U(n)$.) Since $|U(p)| = p - 1$ then by Corollary 18.4(b) $a^{p-1} \equiv 1 \pmod{p}$, therefore $a^p \equiv a \pmod{p}$. □

For example, without doing any calculation we know that $2011^{13} - 2011$ is divisible by 13.

Corollary 18.6 (Euler's Theorem): *Let* $a \in \mathbb{Z}$, $n \in \mathbb{Z}_+$ *and* $\gcd(a,n) = 1$. *Then*

$$a^{\phi(n)} \equiv 1 \pmod{n}.$$

Proof: It suffices to prove the result for $0 < a < n$, since $a^k \equiv r^k \pmod{n}$ for any $k \in \mathbb{N}$, where r is the remainder of a when divided by r. Since $\gcd(a,n) = 1$ then $a \in U(n)$, the multiplicative group of units modulo n. Since $|U(n)| = \phi(n)$ (Euler's ϕ-function) then by Corollary 18.4(b) it follows that

$$a^{\phi(n)} = a^{|U(n)|} \equiv 1 \pmod{n}.$$

□

18.3 Exercises

1. Consider the group $\mathbb{Z}_{12}$ and the subgroup $H = \langle 4 \rangle = \{0, 4, 8\}$.

   (a) Are the following pairs of elements related under $\sim_H$?

   (i) $3, 7$ (ii) $5, 11$ (iii) $6, 9$

   (b) Find all (left) cosets of H in G.

2. In S_7, are the following pairs of elements related under $\sim_H$ where $H = A_7$?

   (a) $(1,2)(3,4)(5,6)$, $(1,7)(2,6)(3,5)(4,7)$.

   (b) $(2,3)(4,6)$, $(1,3,5,7,4)$.

   (c) $(1,3,7,2)$, $(2,4,3,6,5)$.

3. Let $H = \{\varepsilon, (1,3)\}$ in S_3.

   (a) Find all the left cosets of H.

   (b) Find all the right cosets of H.

4. Find all of the left cosets of $H = \{1, 11\}$ in $U(30)$.

5. Let H and K be subgroups of a group G such that $\gcd(|H|, |K|) = 1$. Show that $|H \cap K| = 1$.

6. Suppose that a has order 15. Find all of the left cosets of $\langle a^5 \rangle$ in $\langle a \rangle$.

7. Let $\text{ord}(a) = 30$. How many left cosets of $\langle a^4 \rangle$ in $\langle a \rangle$ are there? List them.

8. Show that the order of $U(n)$ is even when $n > 2$.

9. Let G be a group such that $|G| = 35$.

   (a) Show that G has at most 5 subgroups of order 7.

   (b) Show that G has at most 7 subgroups of order 5.

   (c) Deduce that G has at least one element of order 5 and at least one element of order 7.

10. Let H be a subgroup of a group G with $|H| = \frac{1}{2}|G|$.

(a) Show that $a \notin H$ implies $G = H \cup aH$.

(b) Show that $a \notin H$ implies $a^n H \neq a^{n+1} H$.

(c) Deduce that every element in G which has odd order is contained in H.

11. (a) How many 3-cycles are there in A_5?

    (b) How many 5-cycles are there in A_5?

    (c) Use Exercise 10 to show that A_5 has no subgroup of order 30.

12. Repeat the argument of Exercise11 (modifying it where appropriate) to show that A_4 has no subgroup of order 6.

13. Compute $5^{15} \pmod 7$ and $7^{13} \pmod{11}$.

Lecture 19

Rubik’s Cube: Beginnings

In this lecture, we summarize Rubik’s Cube terminology and notation that we have been using so far, as well as introduce Singmaster notation for each piece of, and each position on, the cube. There is no ”one size fits all” notation when modelling Rubik’s Cube, we’ll see that each notation has its benefits depending on what you are trying to do with it.

19.1 Rubik’s Cube terminology and notation

The notation we use was first introduced by David Singmaster in the early 1980’s, and is the most popular notation in use today.

19.1.1 Move Notation

Fix an orientation of the cube in space. We may label the 6 sides as $\boldsymbol{f}$, $\boldsymbol{b}$, $\boldsymbol{r}$, $\boldsymbol{l}$, $\boldsymbol{u}$, $\boldsymbol{d}$ for *front*, *back*, *right*, *left*, *up*, and *down*.

Face moves:
A quarter twist of a face by 90 degrees in the clockwise direction (looking at the face straight on) is denoted by the uppercase letter corresponding to the name of the face. For example, F denotes the move which rotates the front face by 90 degrees clockwise. See Table 19.1 for a complete description of cube moves and notation.

Slice moves:
We also give names to the **slice** moves. These are moves in which one of the three middle slices is rotated. For example, if the slice between the l and r face is rotated upwards, that is, in the clockwise direction when viewed from the right face, then we denote this move by M_R. We could also view this move from the left side as a counterclockwise rotation, so we could denote it by M_L^{-1}. Similarly, we have slice moves for the slice parallel to the u and d face, and for the slice parallel to the f and b face. These moves are denoted by:

$$M_R = M_L^{-1}, \quad M_U = M_D^{-1}, \quad M_F = M_B^{-1}.$$

See Figure 19.1.

Whole cube moves:
The whole cube, as a single object, can be rotated in space. For example, we can rotate the cube about an axis through the centres of the left and right faces. If the rotation is in the clockwise direction as viewed

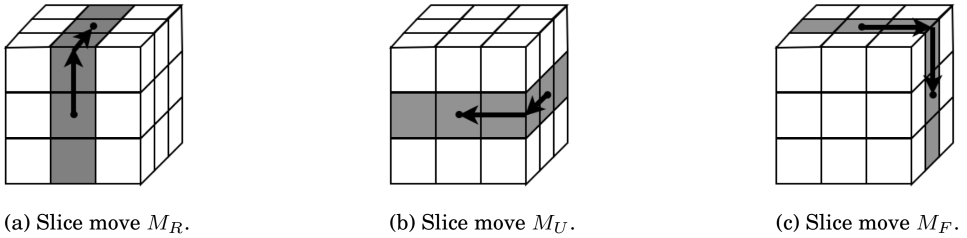

(a) Slice move M_R.
(b) Slice move M_U.
(c) Slice move M_F.

Figure 19.1: Three basic *slice moves* of Rubik's Cube.

from the right face then we denote the move by $\mathcal{R}$. This could also be viewed as a counterclockwise rotation from the left face perspective, so we could also denote it by $\mathcal{L}^{-1}$. See Figure 19.2.

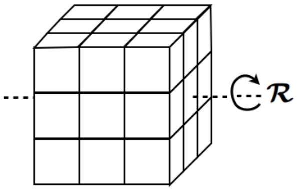


Figure 19.2: Whole cube rotation $\mathcal{R}$. Also denoted by $\mathcal{L}^{-1}$.

19.1.2 Position and Piece Notation

The 26 pieces of the cube, called **cubies**, split up into three distinct types: **centre cubies** (having only coloured sticker), **edge cubies** (having two coloured stickers), **corner cubies** (having three coloured stickers).

We call the space which a cubie can occupy a **cubicle**, and we call the space a sticker can occupy a **facet**. We can also describe a facet as the face of a cubicle. As the pieces move around, the cubies move from cubicle to cubicle, and the stickers move from facet to facet. In the 15-puzzle, Oval Track, and Hungarian Rings puzzles, we called the location a piece could occupy a *position* or *spot*, the terms *cubicle* and *facet* are customary to use when talking about the Rubik's Cube.

To solve the puzzle each cubie must get restored to its original cubicle, we call this the cubies **home location**, and each sticker must get returned to its original facet (i.e. the facets must also be correctly positioned), we call this the cubies **home orientation**.[1] See Figure 19.3 for an example of this distinction. Once *all* cubies are in their home locations and home orientations the puzzle will be solved.

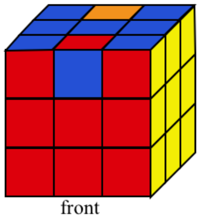


Figure 19.3: Cubie UF is in its home location, but not in its home orientation since it is flipped. Similarly for cubie UB.

[1] This can also be called its **home position.**

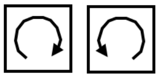
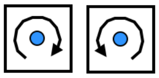
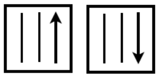
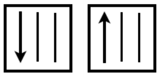
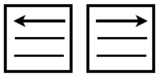
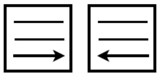
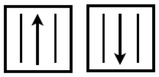
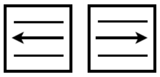
notation	**pictorial**	**description of basic move**
(Singmaster)	**(view from front)**	**(clockwise/counterclockwise refers to viewing the face straight-on)**
F , F^{-1}		F = quarter turn of **front** face in the **clockwise** direction. F^{-1} = quarter turn of **front** face in the **counterclockwise** direction.
B , B^{-1}		B = quarter turn of **back** face in the **clockwise** direction. B^{-1} = quarter turn of **back** face in the **counterclockwise** direction.
R , R^{-1}		R = quarter turn of **right** face in the **clockwise** direction. R^{-1} = quarter turn of **right** face in the **counterclockwise** direction.
L , L^{-1}		L = quarter turn of **left** face in the **clockwise** direction. L^{-1} = quarter turn of **left** face in the **counterclockwise** direction.
U , U^{-1}		U = quarter turn of **up** face in the **clockwise** direction. U^{-1} = quarter turn of **up** face in the **counterclockwise** direction.
D , D^{-1}		D = quarter turn of **down** face in the **clockwise** direction. D^{-1} = quarter turn of **down** face in the **counterclockwise** direction.
M_R , M_R^{-1}		M_R = quarter turn of **vertical** slice in the **clockwise** direction. M_R^{-1} = quarter turn of **vertical** slice in the **counterclockwise** direction.
M_U , M_U^{-1}		M_R = quarter turn of **horizontal** slice in the **clockwise** direction. M_R^{-1} = quarter turn of **horizontal** slice in the **counterclockwise** direction.
$F^2, B^2, R^2, L^2, U^2, D^2$ denote the corresponding *half-turn* of the face. Since a clockwise half-turn is equivalent to a counterclockwise half-turn then $F^2 = F^{-2}, \quad B = B^{-2}, \quad R^2 = R^{-2}, \quad L^2 = L^{-2}, \quad U^2 = U^{-2}, \quad D = D^{-2}$		
$\mathcal{F}, \mathcal{B}, \mathcal{R}, \mathcal{L}, \mathcal{U}, \mathcal{D}$ denote clockwise rotations of the whole cube behind the indicated face.		

Table 19.1: Summary of cube move notation

We will describe a labeling of facets and cubicles below. It is important to keep in mind that facets and cubicles don't move, only the pieces (cubies and stickers) move. So when describing a labeling of the cubicles and facets it is best to think of this label as appearing on a fictitious layer of skin surrounding the puzzle. The pieces can move around under the skin but the skin remains in place.

Cubicle notation:
A cubicle can be identified by the faces it touches. For example, the cubicle that touches the *up*, *right* and *front* faces can be denoted by urf. There was nothing special in how we chose to list these letters, we could denote this cubicle by any one of the 6 symbols: fur, urf, rfu, fru, rfu, or ufr since each gives enough information to describe the *up-right-front* corner cubical. However, the general convention is to use one of the first three symbols fur, urf, or rfu since these list the faces this cubical touches in clockwise order.

Since a corner cubicle has three facets we denote it by three letters. Similarly, edge cubicles are denoted by two letters. Figures 19.4 and 19.5 shows a labeling of all the cubicles (use any one of the facet labelings to denote the cubicle to which the facet belongs).

Facet Notation

Figures 19.4 and 19.5 shows a labeling of the facets of the cube. This labeling is due to mathematician, and puzzle enthusiast, David Singmaster. Our typical labeling uses numbers (see Figure 1.16 in Lecture 1), but this labeling uses strings of symbols. That advantage to this labeling is that it allows us to easily determine where a facet position is located. For example, thinking back to our numerical labeling, if asked where facet 41 is, you likely don't know without looking at a diagram. However, with this new labeling, facet 41 is facet dlf, which you know is on the dlf cubicle. As for which of the three sides it is, this is denoted by the first letter in the name: d for *down*. So facet dlf is the *down* side of the dlf cubicle.

If you are wondering how the order of the other two letters were chosen (i.e. why didn't we call it dfl?), the answer is simple: we wrote them in the order the faces appear when moving around the corner in the clockwise direction. You can check all the labelings in Figure 19.4 to verify this is the convention.

There is a benefit to labeling cubicles and facets in a similar fashion. For the moment we focus our attention on cubies rather than cubicles/facets. For example consider the move R^{-1}. The cubie in cubicle urf moves to cubicle dfr. However, there are three different ways a corner cubie can be placed in a cubicle, so just stating that urf moves to dfr doesn't indicate how it is oriented once it gets to dfr. Notice that the *up* face of the cubie is placed in the *front* face when it moves to the new cubicle. Similarly, the *right* face stays on the *right* face. It would be more descriptive to say that R^{-1} takes the cubie in position urf to position frd. We can write

$$ufr \xrightarrow{R^{-1}} fdr.$$

This indicates that cubie in cubicle ufr moves to cubicle fdr, and the stickers moved as follows: the sticker in the u-facet moves to f-facet, f-facet moves to d-facet, and r-facet moves to r-facet.

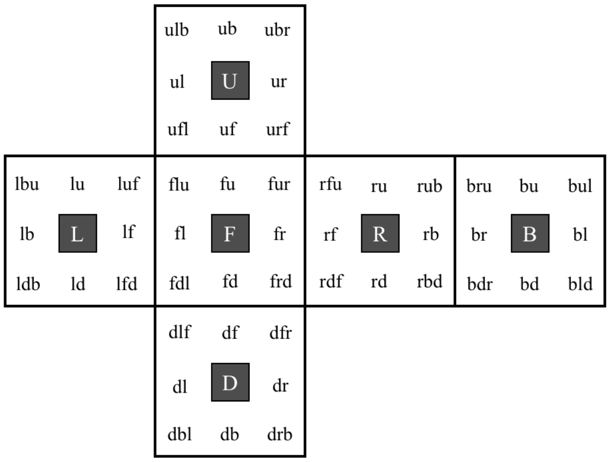
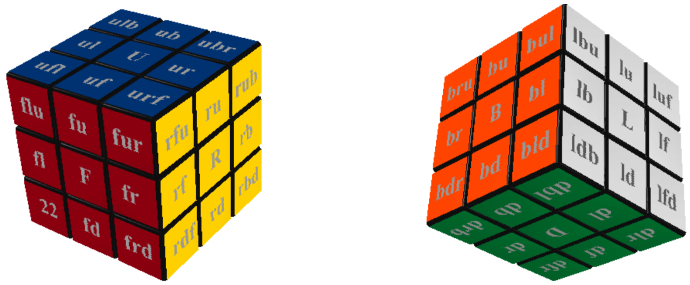

ulb ub ubr
ul U ur
ufl uf urf
lbu lu luf flu fu fur rfu ru rub bru bu bul
lb L lf fl F fr rf R rb br B bl
ldb ld lfd fdl fd frd rdf rd rbd bdr bd bld
dlf df dfr
dl D dr
dbl db drb

Figure 19.4: Facet labeling on the $3 \times 3 \times 3$ Rubik's Cube.



(a) View of front (red), right (yellow) and up (blue) faces, labelled with Singmaster notation.

(b) View of back (orange), left (white) and down (green) faces, labeled with Singmaster notation.

Figure 19.5: Rubik's Cube with classic colouring scheme: blue opposite green, red opposite orange, white opposite yellow. Each cubicle is labeled using Singmaster notation.

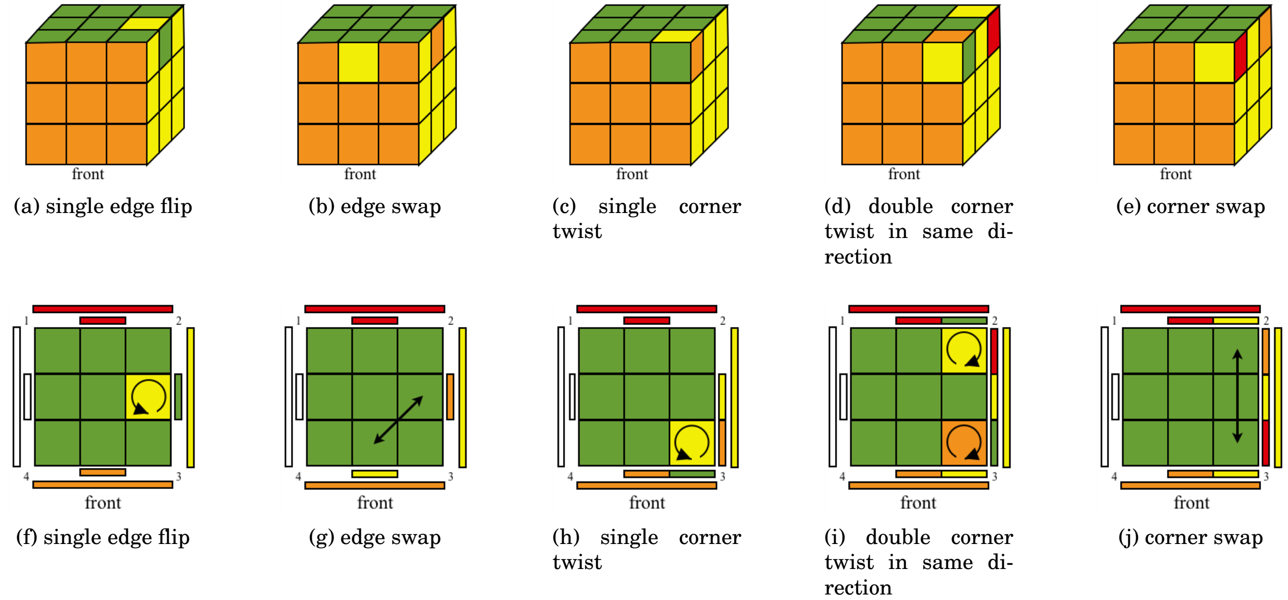
Cubie notation:
A cubie is identified by its home cubicle. We use capital letters to denote cubies, and lower case to denote cubicles. For example, URD denotes the cubie whose home location is the urd cubicle. It may seem that using the same notation to denote cubies as cube moves is a bad idea, however, we'll see that this doesn't cause any trouble at all. We just need to be aware as to whether we are talking about cube moves, or cube pieces.

Table 1.2 in Lecture 1 summarizes the terminology.

With all this notation now in our tool box, we are ready to begin a thorough investigation of Rubik's Cube.

19.2 Impossible Moves

Through previous investigations we've found that there are some moves that are impossible to do on the cube. Figure 19.6 shows five moves that are impossible. This will be helpful when coming up with a strategy to solve the cube since knowing what is impossible to do, will prevent us from going on a search we would never come back from.



(a) single edge flip

(b) edge swap

(c) single corner twist

(d) double corner twist in same direction

(e) corner swap

(f) single edge flip

(g) edge swap

(h) single corner twist

(i) double corner twist in same direction

(j) corner swap

Figure 19.6: Five different moves that are impossible to perform. The image in the bottom row is a face-on perspective of the top face of the corresponding cube in the top row. The thin rectangular boxes on the sides indicate the colour of the side facets, and the long rectangular box indicates the side face colour.

Exercise 14 of Lecture 7 uses permutation-parity arguments to show why it is impossible to (i) flip an edge, (ii) swap two edges, and (iii) swap two corners. The impossibility of the corner twist configurations were investigated using Sage in Exercises 1, 2 and 4 of Lecture 12. In Lecture 20 we will come back to these configurations and give mathematical proofs that they are indeed impossible, thereby confirming the computations done by Sage. Since we were relying on group theoretic algorithms in Sage that are beyond the scope of this book, providing an independent proof will provide us with some closure on this topic.

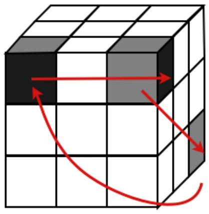
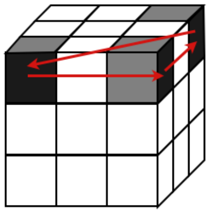
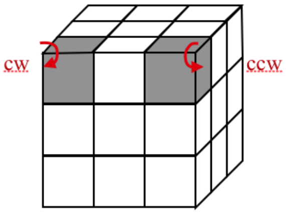
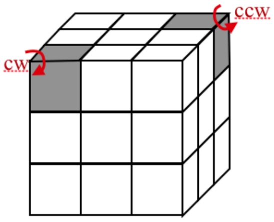
19.3 A Catalog of Useful Move Sequences

Over the previous few lectures we have built some useful moves using commutators. These were move sequences that affected only a few pieces, while returning everything other piece to the position it started. Using conjugation we are able to modify these move sequences to produce other useful moves of the same form. Below is a list of the moves we've created for convenient reference.

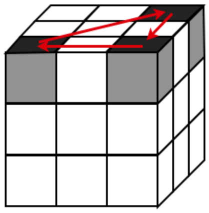
Notice that for each type of cubie (corners and edges) we can (i) 3-cycle any three cubies of the same type, and (b) twist/flip a pair of cubies of the same type. Knowledge of these moves is enough to solve the cube: first place cubies in their home locations (using 3-cycles), then orient the cubies in their home orientation (using twist/flip moves).

Reminder: $[x, y] = xyx^{-1}y^{-1}$ is the *commutator* of x and y and $y^{-1}xy$ is the *conjugate* of x by y. In the following tables, the move labeled C/E# is created using commutators, and the corresponding move denoted by C/E#' is the conjugate of it by the indicated move sequence y.

19.3.1 Corner Moves

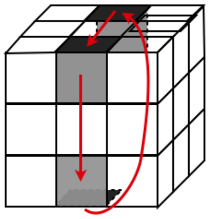
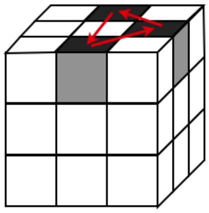
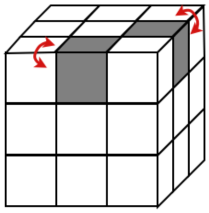
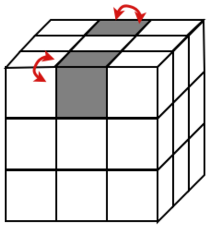
name	effect	move-sequence
C1		$[LD^2L^{-1}, U]$ $= LD^2L^{-1}ULD^2L^{-1}U^{-1}$
C1'		conjugate C1 by $y = B$: $B^{-1}[LD^2L^{-1}, U]B$ $= B^{-1}LD^2L^{-1}ULD^2L^{-1}U^{-1}B$
C2		$[LD^2L^{-1}F^{-1}D^2F, U]$ $= LD^2L^{-1}F^{-1}D^2FUF^{-1}D^2FLD^2L^{-1}U^{-1}$
C2'		conjugate C3 by $y = R$: $R[LD^2L^{-1}F^{-1}D^2F, U]R^{-1}$ $= R^{-1}LD^2L^{-1}F^{-1}D^2FUF^{-1}D^2FLD^2L^{-1}U^{-1}R$

It may be worth noting that we also found the following corner 3-cycle that preserves orientation of the cubies:



The move sequence is $[F^{-1}D^{-1}FR^{-1}D^2RF^{-1}DF, U]$.

19.3.2 Edge Moves

name	effect	move-sequence
E1		$[M_R, U^2]$ $= M_R U^2 M_R^{-1} U^2$
E1'		conjugate E1 by $y = DR^2$: $R^2 D^{-1} [M_R, U^2] D R^2$
E2		$[M_R^{-1} D M_R D^{-1} M_R^{-1} D^2 M_R, U]$
E2'		conjugate E2 by by $y = B^{-1}R^{-1}$: $RB[M_R^{-1} D M_R D^{-1} M_R^{-1} D^2 M_R, U] B^{-1} R^{-1}$

19.4 Strategy for Solution

Our primary goal is in understanding the cube. With that goal in mind we should come away with a strategy for solving the cube. We will not find an optimal strategy, nor will will look for a large collection of moves to tackle all sorts of configurations. Instead, we will be content with a method that systematically solves the cube and uses the tools we have developed in this course. Ideally the method should not involve lots of memorization, but should rely on a solid understanding of the mathematics of permutations, i.e. commutators and conjugates.

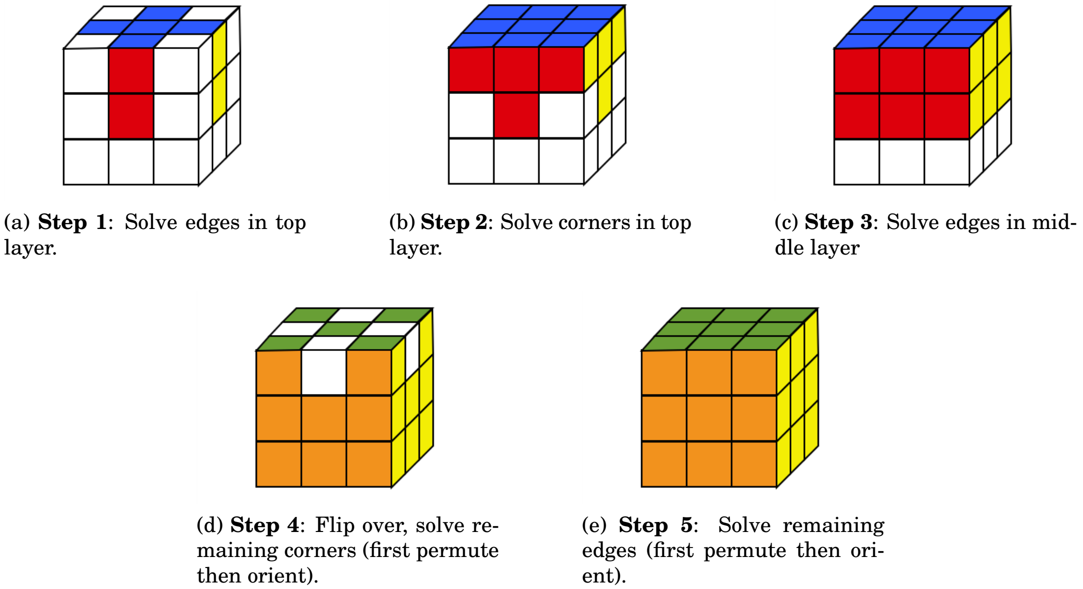
If you haven't already tried to use the moves listed in Section 19.3 to find a strategy yourself, try it now. The fun of discovering a solution on your own may be lost if you read the strategy described below.

More efficient methods than the ones described here, all of which require memorization, are left for the reader to find. A simple google search can keep you busy for weeks.

19.4.1 The Layer Method

The method we will use to solve the cube is known as the *layer method*. We begin by solving the top layer, followed by the middle layer, and finally the bottom layer. A sketch of the steps involved in implementing this strategy are shown in Figure 19.7.

You may begin by solving any colour, and it is best to choose a colour that stands out to you from the rest. This way it is easy to find the pieces on the scrambled cube. In these notes we'll begin by solving the blue layer, in which case the bottom layer will be green.



(a) **Step 1**: Solve edges in top layer.

(b) **Step 2**: Solve corners in top layer.

(c) **Step 3**: Solve edges in middle layer

(d) **Step 4**: Flip over, solve remaining corners (first permute then orient).

(e) **Step 5**: Solve remaining edges (first permute then orient).

Figure 19.7: The Five-Step strategy for solving Rubik's Cube.

Solving the top and middle layers are pretty straightforward. You should be able to do this with a little practice and using general heuristics. A theory based strategy won't be needed until the end-game, which is when we reach the bottom layer.

19.4.2 Solving the Top Layer

Solving the top layer is a straightforward task. You can send pieces to the bottom layer, then bring them back to the top layer, to achieve desired twists. That is, make use of conjugation.

Step 1: Solve the edge cubies in the top layer.

Keep in mind that centres remain fixed, so there is only one proper home orientation for an edge cube. Use the centres as a guide. This is indicated in Figure 19.7a where the centres are shown, and the facets of the edge cubies must match the centres.

Step 2: Solve the corner cubies in the top layer.

Let α be any of the moves R, L, F, B. This will bring one corner cubie into the down layer. Rotating the down layer will then bring a new cubie into the cubicle whose contents are moved back up to the top layer by α^{-1}. In other words, $\alpha D\alpha^{-1}$ allows you to change a corner cubie in the top layer without affecting any other cubies in the top layer. This should help you finish the top layer completely.

19.4.3 Solving the Middle Layer

Step 3: Solve the edge cubies in the middle layer.

We could modify some of the move sequences in Section 19.3 to solve edges in the middle layer. However, this may be overkill, since at this stage there is plenty of "wiggle room" in the down layer so we should be able to find a general heuristic that works. Try to find one yourself.

One method that is pretty straightforward is described here.

If the cubie that is to be placed in the middle layer is currently in the bottom layer then rotate the bottom layer so one sticker of the edge cubie is directly beneath the centre cubie of the same colour. For example, see Figure 19.8 where the cubie to be moved in the middle layer has a red sticker on the side layer, so it is placed directly under the red center cubie. Whatever the colour of your cubie is, rotate the entire cube so the cubie is now in the fd cubicle, and the colour of the sticker in the f face matches the centre cubie of the f face right above it.

Depending on whether the cubie is to be moved to the right of the left we can apply one of the two sequences:

$$\begin{aligned} \text{right:} \quad & [D^{-1}, R^{-1}][D, F] = D^{-1}R^{-1}DRDFD^{-1}F^{-1} \\ \text{left:} \quad & [D, L][D^{-1}, F^{-1}] = DLD^{-1}L^{-1}D^{-1}F^{-1}DF \end{aligned}$$

Notice that in either case the move sequence is a product of Z commutators (see Lecture 13 for a discussion of these commutators).

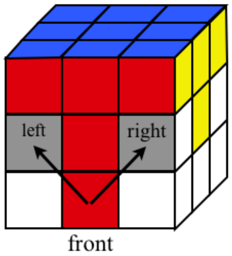


Figure 19.8: Moving an edge piece into the middle layer. To move right apply $[D^{-1}, R][D, F]$, to move left apply $[D, L][D^{-1}, F^{-1}]$.

19.4.4 Solving the Bottom Layer

We now have one layer left to solve. This is the end-game of Rubik's Cube since it is here where things get a bit more difficult. Trying to place the remaining few pieces while leaving previously placed pieces alone requires a collection of strategic moves: ones that move only a few pieces at a time. Luckily, the theory of commutators and conjugates has provided us with such moves (see Section 19.3).

Flip the cube over, so the bottom layer is now the top layer. This will give us clear visibility of all the pieces in the last layer that need to be solved.

Step 4: Solve the remaining corner cubies.

We'll do this in two steps:

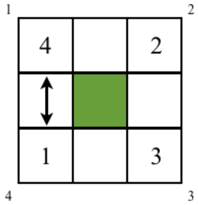
Step 4a: Place the remaining corner cubies in their home locations. Don't worry about twisting them into their home orientations just yet.

Look at the stickers of each of the remaining corner cubies. The colours that appear will tell you exactly where its home location is. For example, the corner cubie with green, white and red stickers belongs to the location which is the intersection of the green, white and red faces. Recall the colour of a face is given by the colour of the centre cubie.

Now that you know where each corner cubie must be moved, see if a simple rotation of the up face will restore all corners to their proper locations. If not, then we are in one of the following cases:

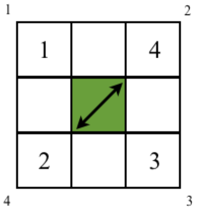
Case 1: It is possible to put exactly one corner cubie in its correct location, and have the other 3 out of position. Use the 3-cycle move sequence C1' in Section 19.3, or its inverse, to move the remaining 3 corner cubies into their correct positions.

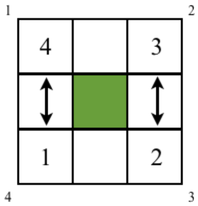
For example, if we need to swap two corner cubies as in the following diagram



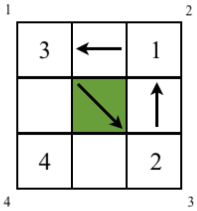
then we can first rotate the face so 1 is home, and 2, 3, 4 are out of position, then we just need to perform a 3-cycle $(2, 4, 3)$.

Case 2: Up to a physical rotation of the whole cube, we are in either one of the two following positions.:





The first case can be taken to the second case by rotating the face counterclockwise $90°$. So assume we are in the second case. Apply C1' to produce the 3-cycle $(1, 4, 2)$, which produces the following position.



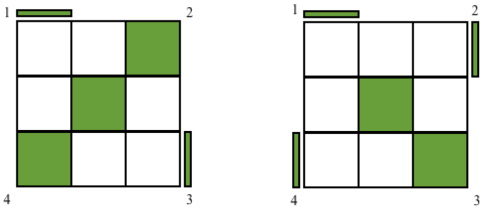
Now use C1' to produce a 3-cycle $(1, 3, 2)$.

Therefore, to restore the corner cubies to their correct locations at most two 3-cycles need to be applied.

Step 4b: Orient (twist) the remaining corner cubies into their home orientations.

Repeated applications of C2 will be enough to orient the corners. (Note, we already used Sage to discover it is impossible to have exactly one corner twisted, or exactly two corners twisted in the same direction.)

Here are a couple of possible scenarios that we could be faced with:



In the first case, applying C2' will solve the corners. In the second case, we can apply $C2'^{-1}$ on corners 1 and 4 to solve corner 4, and twist corner 1 so that it is now out of home position by a clockwise twist. Then applying C2 to corners 1 and 2 will solve the remaining two corners. Other scenarios are possible and can be dealt with similarly.

Step 5: Solve the remaining edge cubies.

We'll do this in two steps:

Step 5a: Place the remaining edge cubies in their home locations. Don't worry about flipping them into their home positions just yet.

Much like the corners, we can use 3-cycles E1' to restore all the edge cubies.

Step 5b: Orient (flip) the remaining edge cubies into their home orientations.

Using E2 and E2' we can flip any pair of edges to restore to their home orientation.

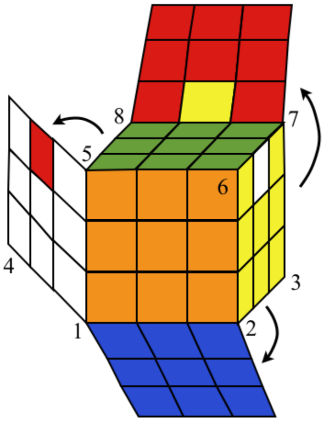
Note, it is impossible to have a single edge flipped as we've already discovered. Therefore, flipped edges occur in pairs and so E2, E2' are the only moves we will need.

Congratulations! Not only have we solved the cube, we built the moves to do it from scratch! Behold the power of the theory of permutations.

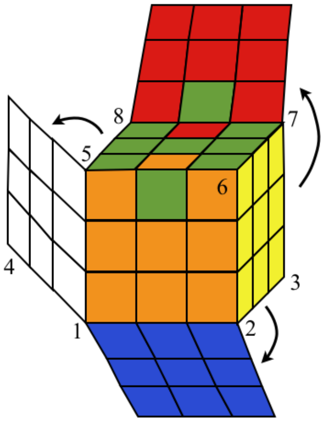
19.5 Exercises

1. Practice solving the first two layers of your cube. Repeatedly scramble and solve until you are confident you can easily solve the first two layers.

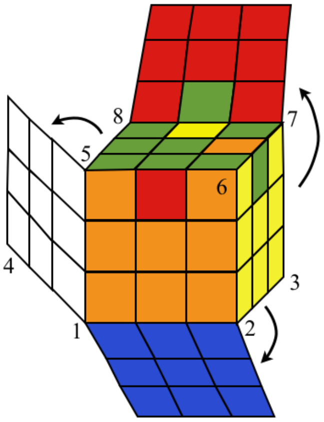
2. **Practice with Step 5: solving edges in final layer.** In each part below, write down a strategy to solve the puzzle.



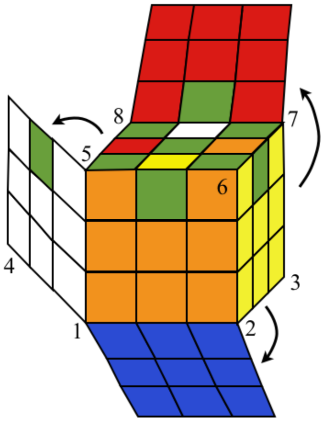
(a)



(b)

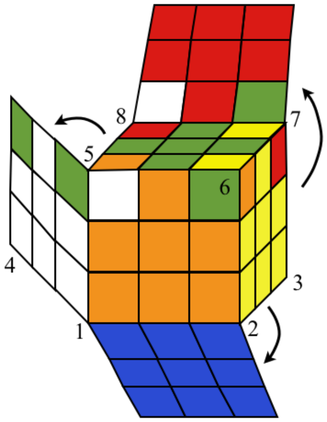


(c)

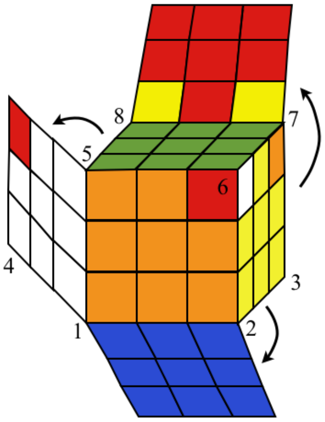


(d)

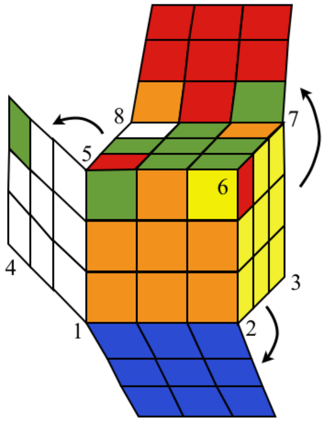
3. **Practice with Step 4: solving corners in final layer.** In each part below, write down a strategy to solve the puzzle.



(a)

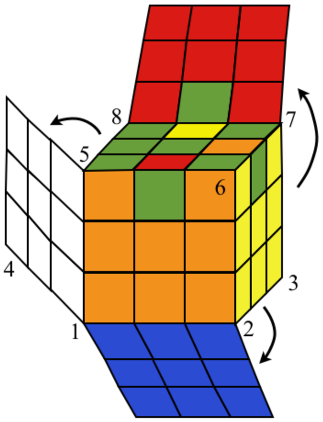


(b)

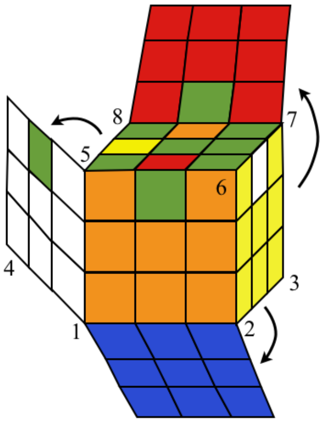


(c)

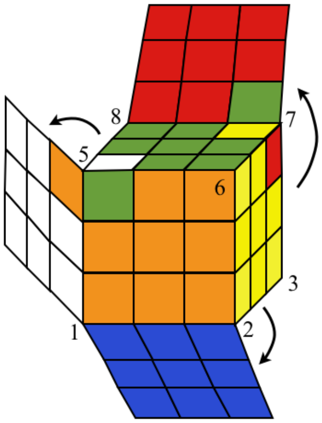
4. **Impossible Configurations.** In each part below, a configuration of the last layer is shown. Show that each configuration is impossible.
(Hint: Try showing the configuration is equivalent to one shown in Section 19.2.)



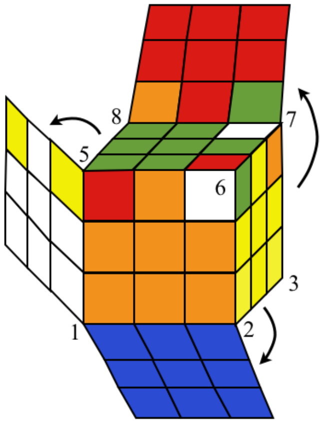
(a)



(b)

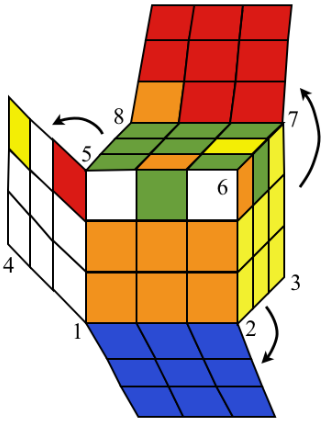


(c)

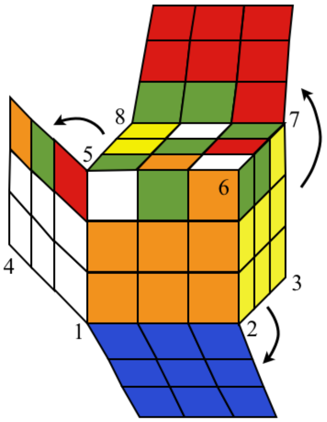


(d)

5. **Practice with Steps 4 and 5: solving corners and edges in final layer.** In each part below, a configuration of the last layer is shown. Write down a strategy to solve the puzzle.



(a)



(b)

Lecture 20

Rubik's Cube: The Fundamental Theorem of Cubology

In this lecture, we present the *Fundamental Theorem of Cubology*. This is the theorem which gives us a complete understanding of what permutations of the cubies are possible, a solvability criteria, and much more.

20.1 Rubik's Cube - A Model

We now describe a mathematical model of Rubik's Cube which is superior to our previous models in a few ways. The difficulty in modeling Rubik's Cube comes from the fact that each cubie has a home location *and* orientation. Sometimes we would like to focus on how the cubies have been *permuted* (without focusing on the orientation of the stickers), and other times we would like to focus on how the cubies are *oriented* in the cubicle they occupy. Our model will consist of a 4-tuple $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$, where ρ (respectively σ) describes how the corner cubies (respectively edge cubies) are permuted, and $\boldsymbol{v}$ (and $\boldsymbol{w}$) describe how the corner cubies (and edges) are oriented.

We begin by fixing an orientation of the cube in space, that is, we choose an up face and a front face. This can be done in any way whatsoever (in fact, there are 24 different ways to do this), but once an orientation is chosen this will remain fixed for the rest of the discussion. In these notes the orientation we will choose is: blue face up, red face in front. We call this the **standard orientation** of the cube. We also assume the classic colouring scheme: blue opposite green, red opposite orange, and yellow opposite white. See Figure 20.1.

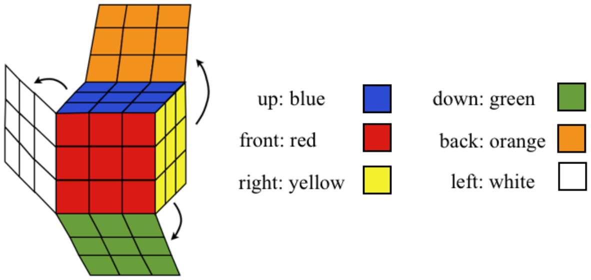
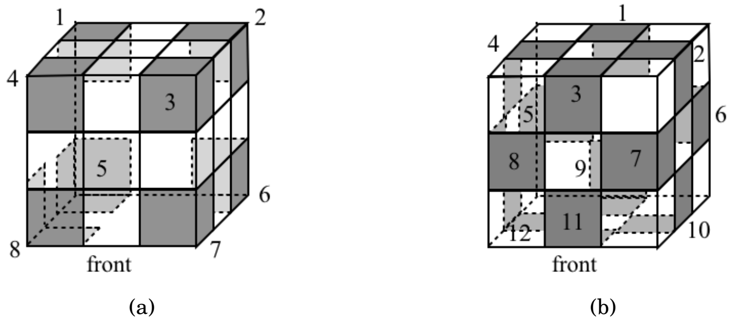


Figure 20.1: The standard orientation for the cube: blue face up, red face front.

We recall some notation:

- V denotes the set of corner cubies. $|V| = 8$.
- E denotes the set of edge cubies. $|E| = 12$.
- RC_3 denotes the Rubik's Cube group.
- $S_V = S_8$ is the symmetric group on the corner cubies.
(Fix a numbering of the corner cubies - see Figure 20.2a.)
- $S_E = S_{12}$ is the symmetric group on the edge cubies.
(Fix a numbering of the edge cubies - see Figure 20.2b.)

As with our other puzzles, we imagine that both the cubies (pieces), and the cubicles (locations), are numbered. When a cubie is in its home location the cubie number will match the cubicle number. Imagine a fictitious layer of skin around the outside of the cube which stays in play under cube moves, the cubicle numbers are printed on this layer of skin. Any configuration of the cube will give two permutations (ignoring orientation of the cubies): $\rho \in S_8$ which corresponds to how the corner cubies are permuted, and $\sigma \in S_{12}$ which corresponds to how the edge cubies are permuted. See Figure 20.2.



(a)

(b)

Figure 20.2: Labeling of the corner and edge cubies.

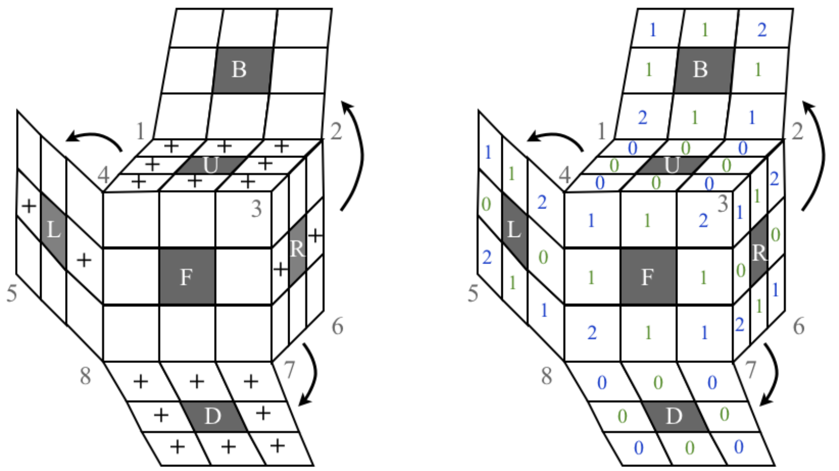
In order to describe the *orientation* of the corner and edge cubies, we mark exactly one facet of each cubicle with a "+" sign. Again, imagine this marking is on the fictitious layer of skin surrounding the cube. Figure 20.3a shows how the facets will be marked. The key thing to observe is that every cubicle has exactly one facet marked. We call this marked facet the *primary facet* of the cubicle.

Next we mark the stickers on each cubie based on their relative position to the primary facet. For this marking, think of the cube in the solved state. For an edge cubie, mark the sticker with a 0 if it is in the primary facet (i.e. beneath the "+" mark on the skin layer), and mark the other sticker on the same cubie with a 1. For a corner cubie, mark the sticker in the primary facet with a 0, and mark the other two stickers with 1 and 2 as you move in the clockwise direction around the cubie. See Figure 20.3b.

For an arbitrary configuration of the cube, the orientation of the edge pieces can be charcterised by a 12-tuple $\boldsymbol{w} = (w_1, w_2, \ldots, w_{12}) \in \mathbb{Z}_2^{12} = \{0, 1\}^{12}$,[1] where w_i is the number on the sticker of the i^{th} edge cubie that is in the primary facet of the cubicle it occupies. Read this last statement again carefully so you know how w_i is defined. Similarly, the orientation of the corner pieces can be charcterised by an 8-tuple $\boldsymbol{v} = (v_1, v_2, \ldots, v_8) \in \mathbb{Z}_3^8 = \{0, 1, 2\}^8$, where v_i is the number on the sticker of the i^{th} corner cubie that is in the primary facet of the cubicle it occupies.

We now have a way to describe the position of all the pieces in any configuration of the cube.

[1] For a set A, A^n denotes the cartesian product of A with itself n times: $A \times A \cdots \times A$.



(a) Marking the primary facets of a cubicle.

(b) Numbering the stickers of a cubie.

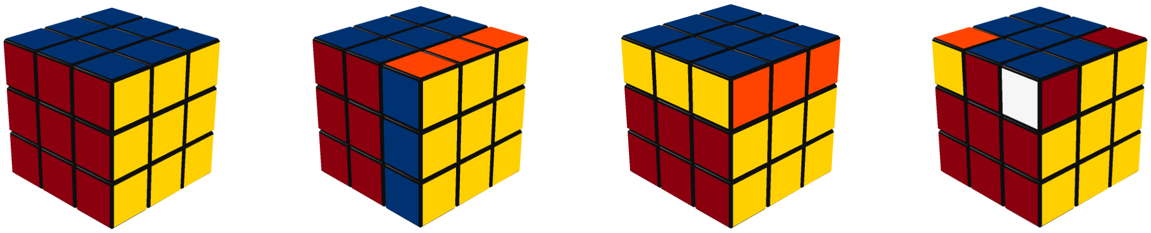
Figure 20.3: Orientation markings.

Definition 20.1 (Position vector of a configuration of cube pieces.): If X is any configuration of Rubik's Cube the **position vector** is a 4-tuple $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ where $\rho \in S_8$, $\sigma \in S_{12}$ encode the permutations of the cubies, and $\boldsymbol{v} \in \mathbb{Z}_3^8$ and $\boldsymbol{w} \in \mathbb{Z}_2^{12}$ encode the orientations of the cubies.

$$\begin{aligned}
\rho \in S_8 :&\quad \rho(i) = j \quad \text{if corner cubie } i \text{ is in cubicle } j.\\
\sigma \in S_{12} :&\quad \sigma(i) = j \quad \text{if edge cubie } i \text{ is in cubicle } j.\\
\boldsymbol{v} = (v_1, v_2, \ldots, v_8) \in \mathbb{Z}_3^8 = \{0,1,2\}^8 :&\quad v_i \text{ is the number on the } i^{\text{th}} \text{ corner cubie beneath}\\
&\quad \text{the “+” mark of the cubicle } \rho(i) \text{ it occupies.}\\
\boldsymbol{w} = (w_1, w_2, \ldots, w_{12}) \in \mathbb{Z}_2^{12} = \{0,1\}^{12} :&\quad w_i \text{ is the number on the } i^{\text{th}} \text{ edge cubie beneath}\\
&\quad \text{the “+” marking of the cubicle } \sigma(i) \text{ it occupies.}
\end{aligned}$$

For simplicity we will use $\mathbf{0}$ to denote the 8-tuple and 12-tuple $(0, 0, \ldots, 0)$.

Let's look at a few examples where we take a configuration and write it as a 4-tuple.



(a) The solved state cube

(b) Move R^{-1}

(c) Move U

(d) Corner 3-cycle C1'

Figure 20.4: Configurations for Example 20.1.

Example 20.1: (a) The solved state cube shown in Figure 20.4a corresponds to the 4-tuple $(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{0})$, since cubies have not been permuted, nor twisted.

(b) Consider the cube corresponding to the move R^{-1} as shown in Figure 20.4b. The corner cubies have been 4-cycled: $\rho = (2, 3, 7, 6)$, and the edge cubies have been 4-cycled: $\sigma = (2, 7, 10, 6)$. To determine the orientation vectors $\boldsymbol{v}$ and $\boldsymbol{w}$, we look at where each cubie was moved to, one by one. Let's start with the corner cubies. Only 4-corner cubies were moved, namely 2, 3, 7 and 6, therefore we only need to figure out what v_2, v_3, v_7 and v_6 are. All others are 0. Cubie 2 (the blue-yellow-orange cubie) has its orange side in the primary facet now, since the orange side is labeled 1 (see brf facet in Figure 20.3) this means $v_2 = 1$. Similarly, cubie 3 (the URF cubie) is now in cubicle frd and the sticker from facet fur (marked with number 2) is now primary facet dfr. Therefore, $v_3 = 2$. The reader should verify the rest of the components in the orientation vectors:

$$\boldsymbol{v} = (0, 1, 2, 0, 0, 2, 1, 0), \qquad \boldsymbol{w} = (0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0).$$

Together with permutations $\rho = (2, 3, 7, 6)$, and $\sigma = (2, 7, 10, 6)$, we have found the 4-tuple position vector.

(c) Consider the cube corresponding to the move U as shown in Figure 20.4c. Since all "+" markings are on the *up*-face each cubie still has the sticker labeled 0 in the primary facet. Therefore,

$$\boldsymbol{v} = \mathbf{0}, \qquad \boldsymbol{w} = \mathbf{0}.$$

The permutations of the cubies are:

$$\rho = (1, 2, 3, 4), \qquad \sigma = (1, 2, 3, 4).$$

(d) Finally, consider the cube corresponding to the move C1' as shown in Figure 20.4d. Edge cubies remained fixed so $\sigma = \varepsilon$ and $\boldsymbol{w} = \mathbf{0}$. The corner cubies are permuted as a 3-cycle $\rho = (2, 4, 3)$ and the orientation vector is:

$$\boldsymbol{v} = (0, 1, 2, 0, 0, 0, 0, 0).$$

Not every 4-tuple $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ corresponds to a legal configuration of Rubik's Cube (i.e. one that is achievable using basic cube moves). For example, the 4-tuple $(\varepsilon, \varepsilon, \mathbf{0}, (1, 0, 0, \ldots, 0))$ represents a single edge flip (where the edge cubie in the ub position was flipped). This is not possible to do through legal cube moves as we have already seen (see Lecture 7 Exercise 14, see also the Sage calculation on page 138). Therefore, the set

$$S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12} = \{(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \mid \rho \in S_8, \sigma \in S_{12}, \boldsymbol{v} \in \mathbb{Z}_3^8, \boldsymbol{w} \in \mathbb{Z}_3^{12}\} \tag{20.1}$$

is much larger than the set of legal cube configurations RC_3. In fact, this set is precisely the set of ways there are to reassemble the cube (assuming you don't take apart the mechanism holding the centres in place, but only disassemble and reassemble edge and corner pieces). We denote set (20.1) by RC_3^* and call it the **illegal cube group** (as opposed to RC_3 which is the (legal) cube group). Previously we used the notation $\mathcal{A}$ to denote this set, but from now on we will use RC_3^* as a reminder of how it is related to RC_3.

Since $RC_3 \subset RC_3^*$ we'd like to characterize exactly which 4-tuples correspond to legal configurations of the cube. This characterization is known as the First Fundamental Theorem of Cubology.

20.2 The First Fundamental Theorem of Cubology

Theorem 20.1 (First Fundamental Theorem of Cubology): *A position vector* $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$ *corresponds to a legal configuration of Rubik's Cube if and only if the following three conditions are satisfied.*

(a) $\textit{sign}(\rho) = \textit{sign}(\sigma)$

(b) $v_1 + v_2 + \cdots + v_8 = 0 \pmod 3$

(c) $w_1 + w_2 + \cdots + w_{12} = 0 \pmod 2$

In words, this theorem says that a configuration is legal if and only if the permutation of the edge cubies has the same parity as the permutation of the corner cubies, the number of clockwise corner twists is equal to the number of counterclockwise corner twists modulo 3, and edge flips occur in pairs.

Verify for yourself that these conditions are satisfied in each case of Example 20.1. Moreover, in the case of a single edge flip in the ub cubicle, the position vector is $(\varepsilon, \varepsilon, \mathbf{0}, (1, 0, 0, \ldots, 0))$ which doesn't satisfy condition (c) of the theorem, hence it isn't a legal configuration.

Proof: (1) First we show that the three conditions are necessary, i.e. that they hold for every legal configuration. To do this we just need to show these conditions hold for the solved state configuration, and they are preserved under the six basic cube moves R, L, U, D, F, B.

The solved state configuration corresponds to the position vector $(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{0})$ and the three conditions in the theorem are satisfied.

For each of the six moves R, L, U, D, F, B the corresponding position vectors are:

$$\begin{aligned}
R &\mapsto ((2,6,7,3),(2,6,10,7),(0,1,2,0,0,2,1,0),(0,1,0,0,0,1,1,0,0,1,0,0)) \\
L &\mapsto ((1,4,8,5),(4,8,12,5),(2,0,0,1,1,0,0,2),(0,0,0,1,1,0,0,1,0,0,0,1)) \\
U &\mapsto ((1,2,3,4),(1,2,3,4),(0,0,0,0,0,0,0,0),(0,0,0,0,0,0,0,0,0,0,0,0)) \\
D &\mapsto ((5,8,7,6),(9,12,11,10),(0,0,0,0,0,0,0,0),(0,0,0,0,0,0,0,0,0,0,0,0)) \\
F &\mapsto ((3,7,8,4),(3,7,11,8),(0,0,1,2,0,0,2,1),(0,0,0,0,0,0,0,0,0,0,0,0)) \\
B &\mapsto ((1,5,6,2),(1,5,9,6),(1,2,0,0,2,1,0,0),(0,0,0,0,0,0,0,0,0,0,0,0))
\end{aligned}$$

and the conditions (a)-(c) of the theorem are satisfied. Each pemutation is a 4-cycle which is odd and has sign -1. The sum of the components of each corner orientation vector is either 0 or 6 which is divisible by 3. The sum of the components of each edge orientation vector is either 0 or 4 which is divisible by 2.

If X is a legal configuration with position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ satisfying (a)-(c) then after applying one of the six basic moves to X (a)-(c) remain satisfied: (a) is satisfied since every one of these moves simultaneously causes a 4-cycle of corners cubies and a 4-cycle of edge cubies, which have the same parity. (b) remains satisfied, because components with U and D don't change at all, while with R, L, F, B simultaneously two components are increased by 1 (modulo 3), and two components are reduced by 1 (modulo 3). (c) remains satisfied, because components with U, D, F, B don't change at all, while with R, L simultaneously two components are increased by 1 (modulo 2), and two components are reduced by 1 (modulo 2).

Since every legal configuration is obtainable from the solved state cube through legal cube moves then properties (a)-(c) are satisfied by any legal configuration.

(2) In order to prove these three conditions are sufficient, we have to show that any position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ satisfying these three properties can be solved using legal cube moves. Our strategy for solving the cube, as laid out in Lecture 19, is enough to prove this part. Let's see why.

Let X be a configuration corresponding to $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$.

(i) Without loss of generality we can assume sign(ρ) = sign$(\sigma) = 1$. If not, just apply any single basic quarter-turn of a face, the resulting position vector will now satisfy this parity condition. This means the permutation of corner cubies is even, and therefore can be restored to their home locations using 3-cycles. Also, edge cubies can be restored to their home locations using 3-cycles. Since we can perform any 3-cycle of corner or edge cubies, then we can restore all cubies to their home locations. Call this new configuration X'.

Since the basic cube moves preserve conditions (a)-(c) then the position vector $(\rho', \sigma', \boldsymbol{v}', \boldsymbol{w}')$ for X' satisfies these conditions, and in this case $\rho' = \varepsilon$, $\sigma' = \varepsilon$. All that remains now is to show we can twist the cubies into their proper orientations.

(ii) Condition (c) says that an even number of edge pieces need to be flipped. Since we have moves to flip any pair of edges then we can solve all the edge cubies. Condition (b) says that the number of clockwise corner twists is equal to the number of counterclockwise corner twists modulo 3. So first twist any cw, ccw pairs into their home orientations. The result will be that all remaining corners twists will occur in triples: 3 cw or 3 ccw twists. These can be solved using our corner twisting moves.

Therefore, X is a solvable configuration. This completes the proof of the theorem. □

The First Fundamental Theorem of Cubology is the solvability criteria for Rubik's Cube. This is the analogue to the solvability criteria that we developed for all the other puzzles. Moreover, this theorem allows us to compute the size of the group RC_3 simply by counting the number of 4-tuples that satisfy the three conditions.

Corollary 20.2 (The Size of the Cube Group): *The number of positions of the illegal and legal Rubik's Cube groups are:*

$$|RC_3| = |\mathcal{C}| = \frac{|RC_3^*|}{12} = 2^{27} \cdot 3^{14} \cdot 5^3 \cdot 7^2 \cdot 11 = 43,252,003,274,489,856,000 \approx 4.3 \cdot 10^{19}.$$
$$|RC_3^*| = |\mathcal{A}| = 8! \cdot 12! \cdot 3^8 \cdot 2^{12}.$$

Proof: Since $RC_3^* = S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$ then $|RC_3^*| = |S_8| \cdot |S_{12}| \cdot |\mathbb{Z}_3|^8 \cdot |\mathbb{Z}_2|^{12} = 8! \cdot 12! \cdot 3^8 \cdot 2^{12}$.

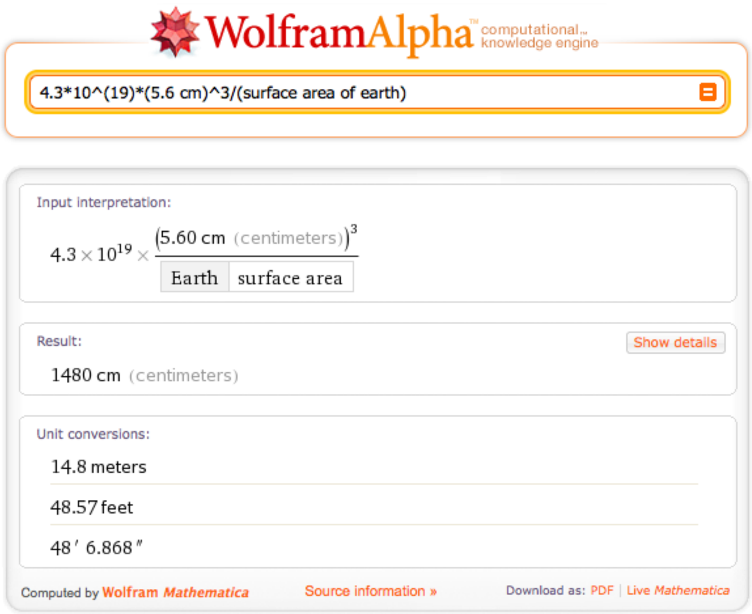
For legal positions this number is reduced by

- half by condition (a) in Theorem 20.1, since there are as many even permutations as there are odd ones,
- a third by condition (b), since the orientation of 7 corner cubies can be arbitrarily chosen and this would determine the orientation of the 8^{th},
- half by condition (c), since the orientation of 11 edge cubies can be arbitrarily chosen and this would determine the orientation of the 12^{th}.

Therefore $|RC_3| = \frac{|RC_3^*|}{12}$. □

How big is this number $|RC_3|$?

If we put $4.3 \cdot 10^{19}$ cubes of 5.6 cm width – each in a different configuration – side by side in a straight line, the length of the line would be $\approx 2.4 \cdot 10^{15}$ kilometres, which is about 255 light years. By way of comparison the star α_1 Centauri is about 4.39 lights years away. Or packed tightly on the surface of the earth the cubes would blanket the earth to a height of 15 metres (see Figure 20.5). Allowing a second for each turn,


WolframAlpha computational knowledge engine
Show details
Computed by Wolfram Mathematica
Source information »
Download as: PDF | Live Mathematica

4.3*10^(19)*(5.6 cm)^3/(surface area of earth)

Input interpretation:

$$4.3 \times 10^{19} \times \frac{(5.60\text{ cm (centimeters)})^3}{\text{Earth} \mid \text{surface area}}$$

Result:

1480 cm (centimeters)

Unit conversions:

14.8 meters

48.57 feet

48′ 6.868″

Figure 20.5: Covering the earth in Rubik's Cubes would create a blanket 15m thick. Calculation on wolframalpha.com.

it would take 1364 billion years to go though all possible configurations (assuming you don't revisit the same configuration twice). By comparison the universe is around 13 billion years old.

Of course it is not the size of the cube group that make Rubik's Cube challenging. After all, if you were given a shuffled deck of 52 playing cards and asked to put them back in order this would be a simple task. Yet there are $52! \approx 8.07 \cdot 10^{67}$ ways the cards could be shuffled, and only one is in the proper order. What makes Rubik's Cube challenging is the way the pieces are linked together, and the restrictions this imposes on legal moves.

20.3 The Second Fundamental Theorem of Cubology

We have two different models for a configuration of Rubik's Cube: (i) the permutation of the 48 faces, as an element in S_{48}, which also corresponds to the move sequence that was applied to the solved state cube to reach the configuration, and (ii) the 4-tuple position vector. The First Fundamental Theorem of Cubology was about the position vector. A restatement of this theorem in terms of move sequences, permutation parity, and twists and flips of cubies is known as The Second Fundamental Theorem of Cubology.

Theorem 20.3 (The Second Fundamental Theorem of Cubology): *A move sequence is possible, if and only if the following three conditions are satisfied:*

(a) *The permutation of the corner and edge cubies (as a product) is even.*

(b) *The number of corners that are twisted clockwise is equal to the number that are twisted counterclockwise modulo 3.*

(c) *The number of flipped edges is even.*

As a consequence we can characterize some impossible move sequences. Notice we have already seen each of these moves to be impossible (some used Sage to investigate these impossibilities). However, now we have given a formal mathematical proof that these are impossible.

Corollary 20.4: *Each of the following configurations cannot be obtained from the solved state cube through legal cube moves.*

(a) Exactly two edge cubies are swapped.

(b) Exactly two corner cubies are swapped.

(c) Exactly one edge cubie is flipped.

(d) Exactly one corner cubie is twisted.

(e) Exactly two corner cubies are twisted in the same direction.

20.4 When are two assembled cubes equivalent?

Consider the equivalence relation $\sim_{RC_3}$ on the illegal cube group RC_3^* defined by:

$$\begin{aligned} X \sim_{RC_3} Y \quad &\iff \quad X^{-1}Y \in RC_3 \\ &\iff \quad X \text{ can be taken to } Y \text{ through a sequence of legal cube moves (i.e. twists of the } 6 \text{ faces).} \end{aligned}$$

All this means is that we consider two assembled cubes equivalent if one can be twisted into the other using legal cube moves.

This partitions RC_3^* into equivalence classes: the left cosets of RC_3 in RC_3^*. The class RC_3 is precisely the set of solvable configurations. We'd like to be able to determine (i) all the other left cosets of RC_3, (ii) a set of representatives for $RC_3^*/\sim_{RC_3}$, and (iii) a quick way to determine to which coset a given cube belongs.

By Corollay 20.2 the number of left cosets is $[RC_3^* : RC_3] = \frac{|RC_3^*|}{|RC_3|} = 12$. The First Fundamental Theorem 20.1 provides a complete characterization of the left cosets. The conditions for a position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ to be in RC_3 are sign$(\rho) =$ sign(σ) and $v_1 + v_2 + \cdots + v_8 = 0 \pmod 3$ and $w_1 + w_2 + \cdots + w_{12} = 0 \pmod 2$. The other cosets are given by the 11 different ways these conditions can fail.

In what follows we will use the notation $X_{(i,j,k)}$, where $i \in \{\pm 1\}$, $j \in \{0,1,2\}$ and $k \in \{0,1\}$, to denote a configuration of the cube where sign$(\rho) \cdot$ sign$(\sigma) = i$, $v_1 + v_2 + \cdots + v_8 = j \pmod 3$, and $w_1 + w_2 + \cdots + w_{12} = k \pmod 2$.

For example, the following conditions on the position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$

$$\text{sign}(\rho) = \text{sign}(\sigma), \qquad v_1 + v_2 + \cdots + v_8 = 0 \pmod 3, \qquad w_1 + w_2 + \cdots + w_{12} = 1 \pmod 2$$

defines the coset $[X_{(1,0,1)}] = X_{(1,0,1)} RC_3$ represented by a single edge flip $X_{(1,0,1)}$ (shown in Figure 20.6c).

$$\text{sign}(\rho) = \text{sign}(\sigma), \qquad v_1 + v_2 + \cdots + v_8 = 1 \pmod 3, \qquad w_1 + w_2 + \cdots + w_{12} = 0 \pmod 2$$

defines the coset $[X_{(1,1,0)}] = X_{(1,1,0)} RC_3$ represented by a single corner twist in the counterclockwise direction $X_{(1,1,0)}$ (shown in Figure 20.6e).

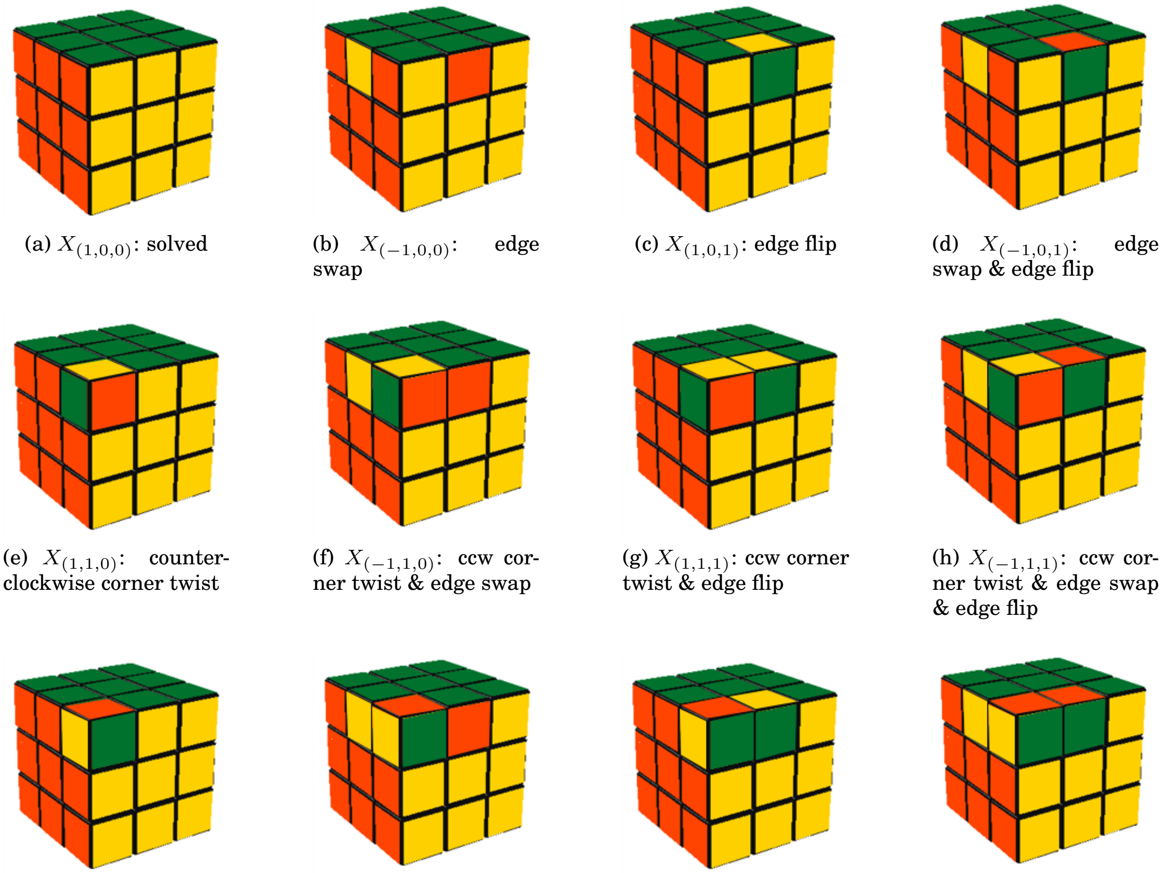
$$\text{sign}(\rho) \neq \text{sign}(\sigma), \qquad v_1 + v_2 + \cdots + v_8 = 0 \pmod 3, \qquad w_1 + w_2 + \cdots + w_{12} = 0 \pmod 2$$

defines the coset $[X_{(-1,0,0)}] = X_{(-1,0,0)} RC_3$ represented by a swap of two edge cubies $X_{(-1,0,0)}$, or equivalently a swap of two corner cubies (shown in Figure 20.6b).

$$\text{sign}(\rho) \neq \text{sign}(\sigma), \qquad v_1 + v_2 + \cdots + v_8 = 2 \pmod 3, \qquad w_1 + w_2 + \cdots + w_{12} = 0 \pmod 2$$

defines the coset $[X_{(-1,2,0)}] = X_{(-1,2,0)}RC_3$ represented by a swap of two edge cubies, and a clockwise twist of a corner cubie $X_{(-1,2,0)}$ (shown in Figure 20.6j). And so on.

Figure 20.6 shows a set of twelve representative for the left cosets of RC_3 in RC_3^*. This means that a randomly assembled cube can be reduced to exactly one of these 12 possibilities.



(a) $X_{(1,0,0)}$: solved

(b) $X_{(-1,0,0)}$: edge swap

(c) $X_{(1,0,1)}$: edge flip

(d) $X_{(-1,0,1)}$: edge swap & edge flip

(e) $X_{(1,1,0)}$: counter-clockwise corner twist

(f) $X_{(-1,1,0)}$: ccw corner twist & edge swap

(g) $X_{(1,1,1)}$: ccw corner twist & edge flip

(h) $X_{(-1,1,1)}$: ccw corner twist & edge swap & edge flip

(i) $X_{(1,2,0)}$: clockwise corner twist

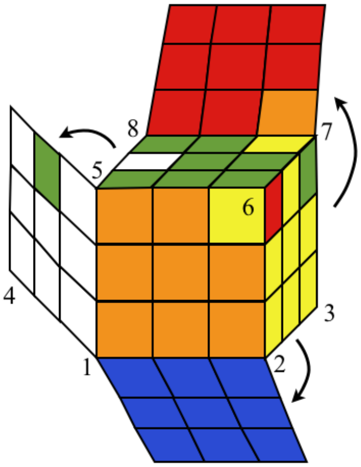
(j) $X_{(-1,2,0)}$: cw corner twist & edge swap

(k) $X_{(1,2,1)}$: cw corner twist & edge flip

(l) $X_{(-1,2,1)}$: cw corner twist & edge swap & edge flip

Figure 20.6: Representatives for the 12 different equivalence classes in $\mathcal{RC}_3^*$

Example 20.2: The following diagram shows a (possibly illegal) configuration of the last layer of Rubik's Cube. We'd like to determine which of the configurations in Figure 20.6 it is equivalent to. To do this it suffices to determine the position vector.



The corners permutation is $\rho = (6, 7)$ and the edge permutation is $\sigma = \varepsilon$. The corner orientation vector is

$$\boldsymbol{v} = (0, 0, 0, 0, 0, 1, 0, 0)$$

since corner cubie 6 is now in position 7 and twisted counterclockwise, and the edge orientation vector is

$$\boldsymbol{w} = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1),$$

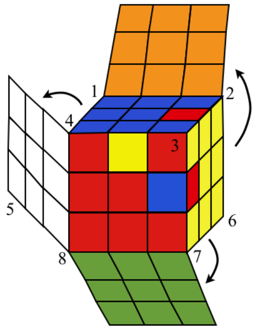
since edge cubie 12 is flipped. Therefore

$$\text{sign}(\rho) \neq \text{sign}(\sigma), \qquad v_1 + v_2 + \cdots + v_8 = 1 \pmod 3 \qquad w_1 + w_2 + \cdots + w_8 = 1 \pmod 2$$

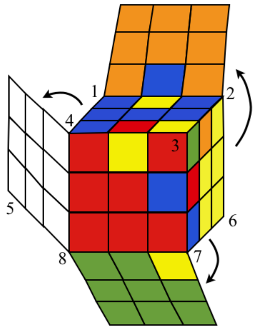
so it is equivalent to the a configuration where: two edges are swapped, one corner is twisted counterclockwise, and one edge is flipped. This is the configuration in Figure 20.6h.

20.5 Exercises

1. For each of the following configurations (i) determine the position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$, and (ii) determine whether it is a legal (i.e. solvable) configuration.
(The corner cubicles are labeled in the diagram below, see Figure 20.2 for a labeling of the edge cubicles.)

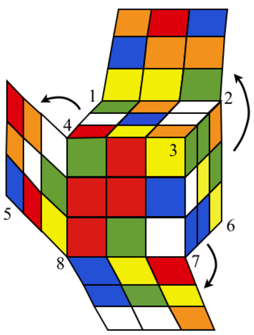


(a)

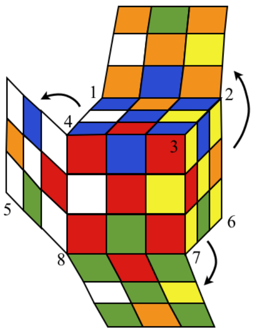


(b)

2. Each of the following configurations is solvable. Determine the position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$ in each case.
(The corner cubicles are labeled in the diagram below, see Figure 20.2 for a labeling of the edge cubicles.)

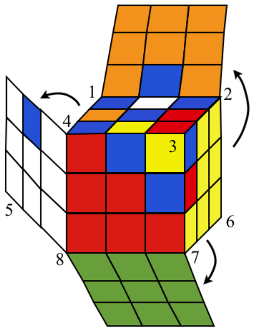


(a)

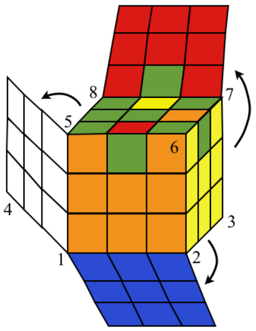


(b)

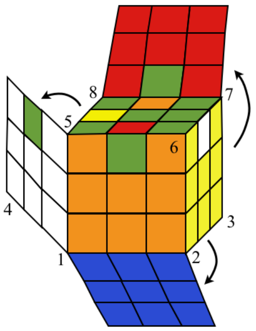
3. Verify the following configuration is not solvable, by showing the position vector doesn't satisfy the three conditions of Theorem 20.1. Determine the quickest way to disassemble/reassemble it so that it becomes solvable. That is, decide if you have to swap two pieces, or flip a single edge, or twist a corner, or a combination of these, etc.



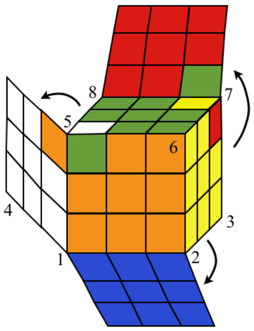
4. **Impossible Configurations.** In each part below, a configuration of the last layer is shown. All non-visible cubies are in their home orientations. Show that each configuration is impossible by showing its position vector doesn't satisfy the three conditions of Theorem 20.1.



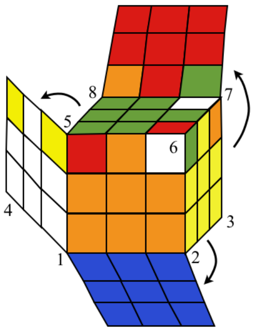
(a)



(b)



(c)



(d)

5. For each of the following move sequences determine the position vector $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$.

   (a) RU

(b) R^2U^2

(c) $(R^2U^2)^3$

(d) $[LD^2L^{-1}, U]$ (a corner 3-cycle)

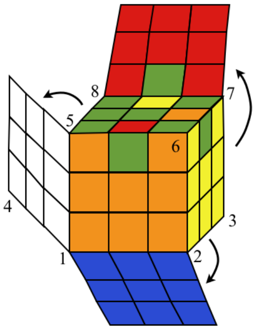
6. For each of the following position vectors $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$ draw the corresponding configuration.
(Assume the standard orientation as shown in Figure 20.1.)
(The puzzles templates file on the webpage includes some Rubik's Cube templates.)

(a) $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) = ((2,4)(1,3), \varepsilon, (1,1,2,2,0,0,0,0), (0,1,1,0,0,0,0,0,0,0,0,0))$

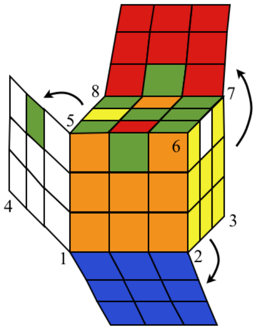
(b) $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) = (\varepsilon, (2,3)(6,7), (0,0,0,0,0,0,0,0), (0,0,0,0,0,0,0,0,0,0,0,0))$

(c) $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) = ((2,4)(3,7), (2,7,3), (0,1,2,1,0,0,2,0), (0,0,1,1,0,0,0,0,0,0,0,0))$

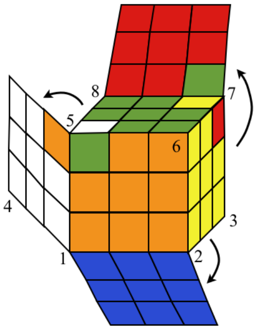
7. In each part below, a configuration Y_i of the last layer for a cube in RC_3^* is shown. Determine the representative $X_{i,j,k}$ (from Figure 20.6) for the coset to which configuration Y_i belongs. That is, determine $i \in \{\pm 1\}, j \in \{0,1,2\}, k \in \{0,1\}$ for which $Y_i \in [X_{i,j,k}] = X_{i,j,k}RC_3$.



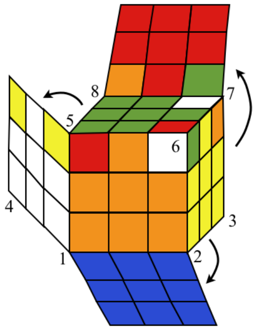
(a) Y_1



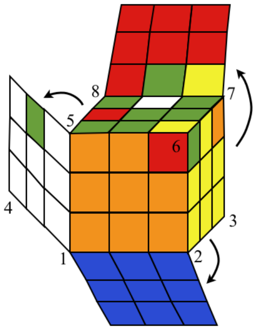
(b) Y_2



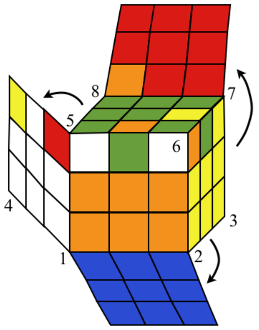
(c) Y_3



(d) Y_4

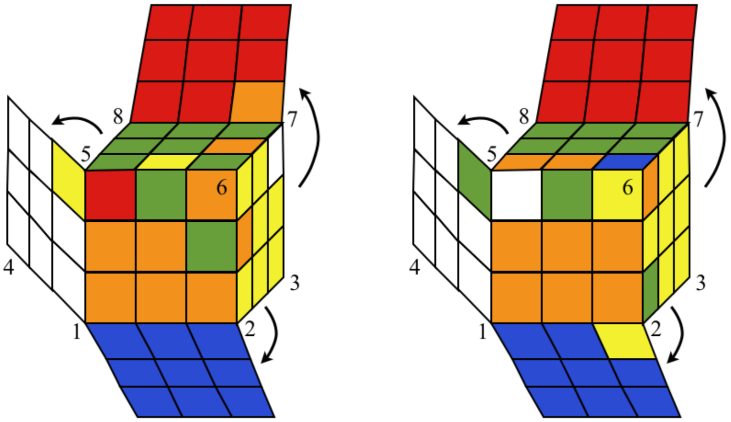


(e) Y_5



(f) Y_6

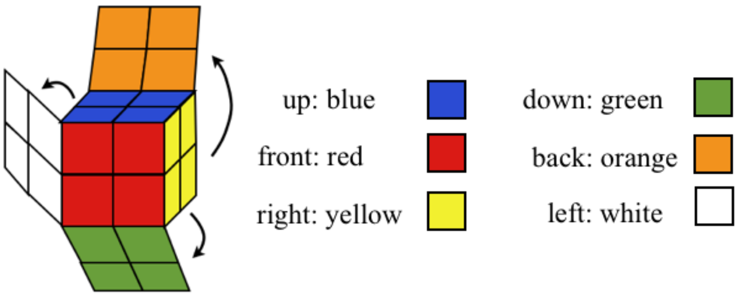
8. Are the following two (possibly illegal) configurations equivalent under cube moves?



(a) X (b) Y

9. **Explorations of the Pocket Cube:** $2 \times 2 \times 2$ **cube.**

The $2 \times 2 \times 2$ Rubik's Cube is shown in the figure.



In this question you are asked to discover an analogous version of the *Fundamental Theorem of Cubology* (Theorem 20.1) as applied to this cube. You are also asked to determine the number of inequivalent illegal cubes, and a set of representatives. This requires a thorough understanding of the material in this lecture since you will have to do a similar analysis for this smaller cube.

To begin, we have to notice that there are no fixed centres for any of the sides. In fact, every cubie can be moved on this puzzle. Therefore, specifying a frame of reference requires some care. Notice that it is sufficient to only use the three moves F, R and D. Since, for example, an L move results in the same configuration as an R move (up to a rotation of the whole cube). Restricting our basic moves to just F, R and D means the ulb cubie remains fixed. We will use this cubie as our frame of reference.

(a) Define a numbering of the cubies and cubicles. (Analogous to Figure 20.2.) Include a picture of your numbering scheme. (Note, you done need to number the cubie in ulb since it doesn't get moved by F, R and D. This means you only need to number the remaining 7 cubies.)

(b) Define what you will mean by the primary facet of each cubicle. (Analogous to Figure 20.3a.) Include a picture of your labeling. (See templates file at [10] for a $2 \times 2 \times 2$ template.)

(c) Assign orientation numbers to each sticker. (Analogous to Figure 20.3b.) Include a picture of your labeling.

(d) Give a definition for a *position vector* of a configuration. This should contain complete information about the configuration.

(e) Give some examples of configurations and the corresponding positions vectors.

(f) Define the terms: *illegal* Pocket Cube groups RC_2^* and *legal* Pocket Cube group RC_2. (Analogous to the $3 \times 3 \times 3$ description on page 234.)

(g) State a *Fundamental Theorem of Pocket Cubology*. (Analogous to Theorem 20.1)

(h) Prove your *Fundamental Theorem of Pocket Cubology*. (See what parts of the proof of Theorem 20.1 can still be used/modified.)

(i) Use your theorem to determine the the size of the legal pocket cube group RC_2. (Analogous to Corollary 20.2)

(j) Determine a condition on the position vector to determine when two assembled pocket cubes in RC_2^* are equivalent under cube moves. Find the number of inequivalent ways there are to assemble a pocket cube, and determine a set of representative for each equivalence class. In other words, do what was done in Section 20.4 but now for the pocket cube.

Emphasis here is on explanations. Write in full sentences and provide clear detailed descriptions.

Lecture 21

Rubik's Cube: Subgroups of the Cube Group

In this lecture, we consider various collections of moves on Rubik's Cube and determine the subgroups they generate. We also see what the Fundamental Theorem of Cubology tells us about the structure of the group operation on RC_3 and we show the only (nontrivial) move sequence that commutes with *every* other move sequence is the *superflip*.

21.1 Building Big Groups from Smaller Ones

Starting with a collection of groups we can stick them together to form a new, larger group.

Given a finite collection of groups $G_1, G_2, \ldots G_n$, the **direct product** of $G_1, G_2, \ldots G_n$ is

$$G_1 \oplus G_2 \cdots \oplus G_n = \{(g_1, g_2, \ldots, g_n) \mid g_i \in G_i\}$$

which is a group under the operation:

$$(g_1, g_2, \ldots, g_n)(h_1, h_2, \ldots, h_n) = (g_1h_1, g_2h_2, \ldots, g_nh_n).$$

It is understood that each product g_ih_i is performed with the operation of group G_i.

To see why $G_1 \oplus G_2 \cdots \oplus G_n$ is a group under this operation we observe:

1) It is closed since each G_i is closed under its operation.

2) The operation is associative since the operations on each of the G_i's is associative.

3) The identity is $(e_1, e_2, \ldots, e_n)$ where each e_i is the identity of G_i.

4) The inverse of an element $(g_1, g_2, \ldots, g_n)$ is $(g_1^{-1}, g_2^{-1}, \ldots, g_n^{-1})$.

Example 21.1: The direct product of S_3 and $\mathbb{Z}_5$ consists of $3! \cdot 5 = 30$ elements. For example $((1,3,2),4)$, and $((1,2),3)$ are two elements in $S_3 \oplus \mathbb{Z}_5$. The product of these elements is

$$((1,3,2),4)\ ((1,2),3) = ((1,3,2)(1,2), 4+3) = ((1,3),2).$$

For simplicity let's just limit our attention to the direct product of two groups: $G \oplus H$. The subset

$$G \oplus \{e_H\} := \{(g, e_H) \mid g \in G\}$$

is a subgroup of $G \oplus H$ which essentially a copy of G. Similarly,

$$\{e_G\} \oplus H := \{(e_G, h) \mid h \in H\}$$

is a subgroup of $G \oplus H$ which essentially a copy of H, In other words, we have used G and H to build a bigger group $G \oplus H$ in which G and H are subgroups.

Example 21.2: The group $\mathbb{Z}_2^3 := \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2$ is a group of order 8, and every non-identity element has order 2.

The group $\mathbb{Z}_2 \oplus \mathbb{Z}_3$ is a cyclic group of order 6, since the element $(1, 1)$ has order 6 (check this).

$$\mathbb{Z}_2 \oplus \mathbb{Z}_3 = \{(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2)\}.$$

For a group G, we denote by G^n the direct product with itself n-times: $G^n = G \oplus G \cdots \oplus G$.

21.2 Some Subgroups of RC_3

In this section we investigate some of the types of groups that appear as subgroups of the Rubik's Cube. In Chemistry, one my be interested in what elements make up a compound. As an analogy, think of the Rubik's Cube group as the "compound", and the "elements" that make it up are the subgroups. We'd like to see what kinds of groups live inside RC_3.

It is particularly interesting to "realize" a finite group A as a subgroup of the cube. This can be done for all groups of order < 13; the smallest abelian group which is not a subgroup of RC_3 is $\mathbb{Z}_{13}$ (since $13 \nmid |RC_3|$ by Corollary 20.2), and the smallest non-abelian group is D_{13}. In the next few sections, we'll see a few examples of some groups that live inside RC_3.

21.2.1 Cyclic subgroups and orders of elements in RC_3

The easiest type of subgroup to look for are the cyclic subgroups. Since the order of an element is precisely the size of the cyclic group it generates then we are really just interested in what are the possible orders of elements in RC_3.

An element of order 4 is R. So RC_3 contains a cyclic group of order 4 as a subgroup: $\mathbb{Z}_4 = \langle R \rangle$.

The move sequence R^2U^2 has order 6, so RC_3 contains as cyclic subgroup of order 6: $\mathbb{Z}_6 = \langle R^2U^2 \rangle$.

The move sequence RU has order 105 and the move sequence RU^{-1} has order 63. Therefore, RC_3 contains copies of $\mathbb{Z}_{63}$ and and $\mathbb{Z}_{105}$ as subroups.

Sage

```
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
sage: RC3=S48.subgroup([R,L,U,D,F,B])
sage: (R*U).order()
105
sage: (R*U^(-1)).order()
63
```

There exist precisely 73 different orders of elements in RC_3 and the maximum order is 1260. The move sequence $RU^2D^{-1}BD^{-1}$ has order 1260. (See page 93 of [7], or page 51 of [2].)

21.2.2 Two Squares Group: $\langle R^2, U^2\rangle$

Let $H = \langle R^2, U^2\rangle$ denote the group generated by the square moves R^2 and U^2. The group contains the useful 2-pair edge swap: $(R^2U^2)^3$.

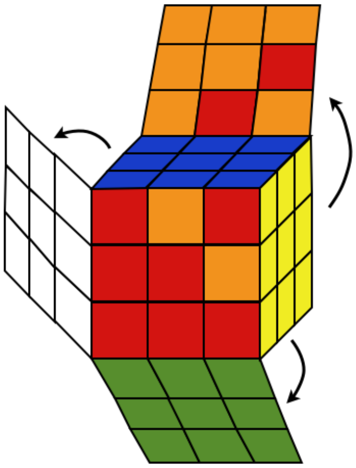


Figure 21.1: The two pair edge swap $(R^2U^2)^3$ in $H = \langle R^2, U^2\rangle$.

We can find all the elements of this group fairly easily:

$$\begin{aligned} H =& \{1, R^2, R^2U^2, R^2U^2R^2, (R^2U^2)^2, (R^2U^2)^2R^2, (R^2U^2)^3, \\ & (R^2U^2)^3R^2, (R^2U^2)^4, (R^2U^2)^4R^2, (R^2U^2)^5, (R^2U^2)^5R^2\}, \end{aligned}$$

Therefore, $|H| = 12$. Note that $1 = (R^2U^2)^6$, $U^2 = (R^2U^2)^5R^2$, and $U^2R^2 = (R^2U^2)^5$.

We can compute the order of each element one by one and see that the maximum order is 6. This can also be done quickly in Sage.

Sage

```
sage: S48=SymmetricGroup(48)
sage: R=S48("(25,27,32,30)(26,29,31,28)(3,38,43,19)(5,36,45,21)(8,33,48,24)")
sage: L=S48("(9,11,16,14)(10,13,15,12)(1,17,41,40)(4,20,44,37)(6,22,46,35)")
sage: U=S48("(1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19)")
sage: D=S48("(41,43,48,46)(42,45,47,44)(14,22,30,38)(15,23,31,39)(16,24,32,40)")
sage: F=S48("(17,19,24,22)(18,21,23,20)(6,25,43,16)(7,28,42,13)(8,30,41,11)")
sage: B=S48("(33,35,40,38)(34,37,39,36)(3,9,46,32)(2,12,47,29)(1,14,48,27)")
sage: RC3=S48.subgroup([R,L,U,D,F,B])
sage: H=S48.subgroup([R^2,U^2])
sage: [g.order() for g in H]
[1, 2, 2, 2, 2, 3, 2, 6, 2, 3, 2, 6]
```

We've just discovered that H is a group of order 12, with two elements of order 6, two elements of order 3, and seven elements of order 2. This seems eerily reminiscent of the dihedral group D_6. Let check to see H is really D_6 in disguise.

Sage

```
sage: H.is_isomorphic(DihedralGroup(6))
True
```

It is! We've just discovered that the dihedral group D_6 lives inside the Rubik's Cube group. [1]

21.2.3 The Slice Squared Group: $\langle M_R^2, M_U^2, M_F^2\rangle$

Let $H = \langle M_R^2, M_U^2, M_F^2\rangle$ denote the group generated by the square slice moves.

Each of the generators M_R^2, M_U^2, M_F^2 has order 2, and each of the products

$$M_R^2 M_F^2, \quad M_R^2 M_U^2, \quad M_F^2 M_U^2$$

has order 2 also (play with your cube to see this). This means that H is an abelian group (Why?), and *every* element has order 2.

For simplicity of notation let $a = M_R^2$, $b = M_F^2$ and $c = M_U^2$ then it is straightforward to see that:

$$H = \{1, a, b, c, ab, ac, bc, abc\},$$

is a group of order 8. In fact, $H \approx \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2$ under the correspondence

$$\begin{aligned} 1 &\leftrightarrow (0,0,0)\\ a &\leftrightarrow (1,0,0)\\ b &\leftrightarrow (0,1,0)\\ c &\leftrightarrow (0,0,1)\\ ab &\leftrightarrow (1,1,0)\\ ac &\leftrightarrow (1,0,1)\\ bc &\leftrightarrow (0,1,1)\\ abc &\leftrightarrow (1,1,1)\end{aligned}$$

See Figure 21.2.

21.3 Structure of the Cube Group RC_3

Let X and Y be two elements of RC_3 with corresponding position vectors $(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})$ and $(\rho^*, \sigma^*, \boldsymbol{v}^*, \boldsymbol{w}^*)$, respectively.

Recall, this notation means that corner cubie i moved to cubicle $\rho(i)$ and v_i is the label on the sticker beneath the primary face labeled "+", and edge cubie i moved to edge cubicle $\sigma(i)$ with label w_i on the sticker in the primary facet labeled "+". If we compose the moves X and Y then the position vector of XY can be obtained as follows:

- corner cubie i moves to $(\rho\rho^*)(i) = \rho^*(\rho(i))$,
- edge cubie i moves to $(\sigma\sigma^*)(i) = \sigma^*(\sigma(i))$,
- the label on the i^{th} corner cubie, in the primary facet of cubicle $(\rho\rho^*)(i)$, is $v_i + v^*_{\rho(i)} \pmod 3$.
- the label on the i^{th} edge cubie, in the primary facet of cubicle $(\sigma\sigma^*)(i)$, is $w_i + w^*_{\sigma(i)} \pmod 2$.

[1] We say two groups G_1 and G_2 are **isomorphic** if they have the same group structure (i.e. same Cayley table), but the names of the elements could be different. More precisely, we mean there is a map $\phi : G_1 \to G_2$ which is a bijection, and for any $g, h \in G_1$, $\phi(gh) = \phi(g)\phi(h)$. Sage has built in functionality for checking whether two groups are really the same (i.e. isomorphic).

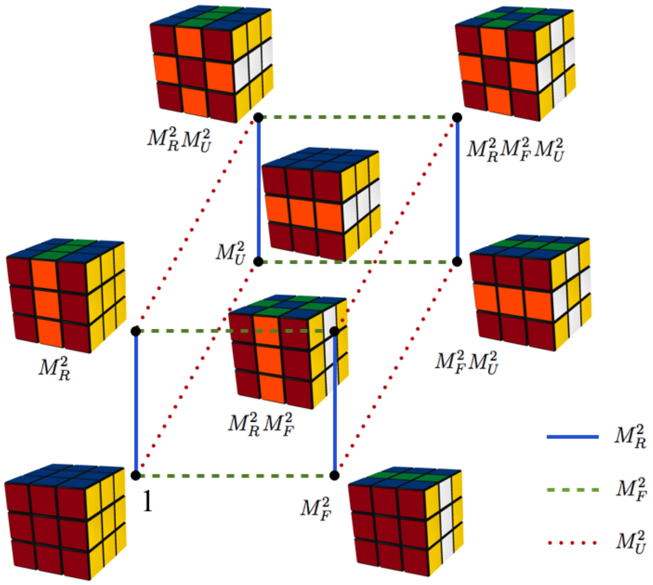


Figure 21.2: Cayley graph of H: The elements in the slice squared group and their representations in terms of the generators.

If we define addition of 8-tuple (and 12-tuple) orientation vectors componentwise: i.e. $\boldsymbol{a}+\boldsymbol{b} = (a_1, a_2, \ldots, a_k) + (b_1, b_2, \ldots, b_k) = (a_1 + b_1, a_2 + b_2, \ldots, a_k + b_k)$ (i.e. think $\mathbb{Z}_3^8 = \mathbb{Z}_3 \oplus \mathbb{Z}_3 \oplus \cdots \oplus \mathbb{Z}_3$ and $\mathbb{Z}_2^{12} = \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \cdots \oplus \mathbb{Z}_2$) then the group operation on $RC_3 = S_8 \times S_{12} \times \mathbb{Z}_3^8 \times \mathbb{Z}_2^{12}$ is:

$$(\rho, \sigma, \boldsymbol{v}, \boldsymbol{w})(\rho^*, \sigma^*, \boldsymbol{v}^*, \boldsymbol{w}^*) = (\rho\rho^*, \sigma\sigma^*, \boldsymbol{v} + \rho(\boldsymbol{v}^*), \boldsymbol{w} + \sigma(\boldsymbol{w}^*)) \tag{21.1}$$

where $\rho(\boldsymbol{v}^*)$ represents the orientation vector obtained from $\boldsymbol{v}^*$ by replacing the i^{th} component v_i with $v_{\rho(i)}$:

$$\rho(\boldsymbol{v}^*) = \rho((v_1^*, v_2^*, \ldots, v_8^*)) = (v_{\rho(1)}^*, v_{\rho(2)}^*, \ldots, v_{\rho(8)}^*).$$

and $\sigma(\boldsymbol{w}^*)$ represents:

$$\sigma(\boldsymbol{w}^*) = \sigma((w_1^*, w_2^*, \ldots, w_{12}^*)) = (w_{\sigma(1)}^*, w_{\sigma(2)}^*, \ldots, w_{\sigma(12)}^*).$$

Let

$$G_1 = \{g = (\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in RC_3 \mid \boldsymbol{v} = \boldsymbol{0}, \boldsymbol{w} = \boldsymbol{0}\}$$
$$G_2 = \{g = (\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in RC_3 \mid \rho = \varepsilon, \sigma = \varepsilon\}.$$

Then G_1 and G_2 are subgroups of RC_3. G_1 is the subgroup of all move sequences which preserves the orientation of all the pieces. G_2 is the subgroup of all move sequences which leaves every cubie in its own cubicle, but may flip/twist the cubies.

The following theorem describes how the subgroups G_1 and G_2 are interlinked in order to form RC_3. Some of the terms are not explained as it is a more advanced theorem. I include it here only for the benefit of those who know about: normal subgroups, isomorphisms, and semidirect products.

Theorem 21.1:

(a) G_1 *is a subgroup,* G_2 *is a normal subgroup of* RC_3. [a]

(b) $G_1 \approx \{(\rho, \sigma) \in S_8 \times S_{12} \mid sign(\rho) = sign(\sigma)\}$, $G_2 \approx \mathbb{Z}_8^7 \times \mathbb{Z}_2^{11}$.

(c) RC_3 *is the semidirect product of* G_1 *with* G_2.

[a] A **normal subgroup** is a subgroup H of a group G with the property that all its left and right cosets are equal: $aH = Ha$ for all $a \in G$. Such subgroups are extremely important in advanced group theory.

21.3.1 The Centre of the Cube group, $Z(RC_3)$, and the Superflip

Recall that for any group G, the **centre** of G, denoted by $Z(G)$ is the set of all elements that commute with every element of G:

$$Z(G) = \{a \in G \mid ag = ga \text{ for all } g \in G\}.$$

The centre is a subgroup of G. (See Section 11.3)

Theorem 21.2: *The centre of* RC_3 *consists of two elements: the identity* ε *and the superflip* X_{SF}. *The superflip, is the configuration in which every cubie is in its home location but all the edge cubies are flipped (see Figure 21.3).*

$$Z(RC_3) = \{\varepsilon, X_{SF}\}.$$

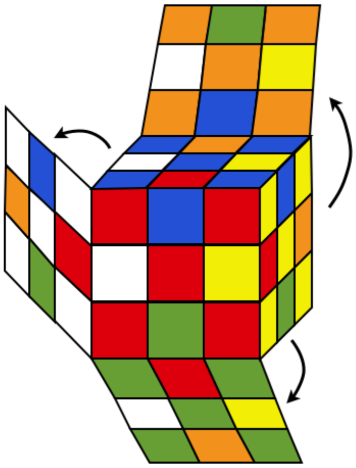


Figure 21.3: The superflip configuration of Rubik's Cube: X_{SF}.

Proof: Let $g = (\rho, \sigma, \boldsymbol{v}, \boldsymbol{w}) \in Z(RC_3)$. Since the centre of the symmetric group S_n, for $n \geq 3$, is trivial and since every $\rho^* \in S_8$ appears as a first coordinate of the position vector, it immediately follows from Equation 21.1 that $\rho = \varepsilon$, and similarly $\sigma = \varepsilon$. That is, $g = (\varepsilon, \varepsilon, \boldsymbol{v}, \boldsymbol{w}) \in G_2$. Thus, $gg^* = g^*g$ simply becomes $\boldsymbol{v} + \boldsymbol{v}^* = \boldsymbol{v}^* + \rho^*(\boldsymbol{v})$, i.e. $\boldsymbol{v} = \rho^*(\boldsymbol{v})$ for all $\rho^* \in S_8$, and $\boldsymbol{w} + \boldsymbol{w}^* = \boldsymbol{w}^* + \sigma^*(\boldsymbol{w})$, i.e. $\boldsymbol{w} = \sigma^*(\boldsymbol{w})$ for all $\sigma^* \in S_{12}$. This means the $\boldsymbol{v}$ and $\boldsymbol{w}$ are constant (i.e. $v_i = v_j$ for all $1 \leq i, j \leq 8$ and $w_i = w_j$ for all $1 \leq i, j \leq 12$). So we have

$$\boldsymbol{v} = (0,0,0,0,0,0,0,0) = \mathbf{0} \quad \text{or} \quad \boldsymbol{v} = (1,1,1,1,1,1,1,1) = \mathbf{1} \quad \text{or} \quad \boldsymbol{v} = (2,2,2,2,2,2,2,2) = \mathbf{2}$$

and

$$\boldsymbol{w} = (0,0,0,0,0,0,0,0,0,0,0,0) = \mathbf{0} \quad \text{or} \quad \boldsymbol{w} = (1,1,1,1,1,1,1,1,1,1,1,1) = \mathbf{1}.$$

The First Fundamental Theorem of Cubology (Theorem 20.1) excludes the cases $\boldsymbol{v} = \mathbf{1}, \mathbf{2}$, therefore $\boldsymbol{v} = \mathbf{0}$. Both choices for $\boldsymbol{w}$ are possible. This means g is either $(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{0})$ or $(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{1})$. Therefore,

$$Z(RC_3) = \{(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{0}), (\varepsilon, \varepsilon, \mathbf{0}, \mathbf{1})\}.$$

The configuration $(\varepsilon, \varepsilon, \mathbf{0}, \mathbf{1})$ is the superflip. □

21.4 Exercises

1. Consider the direct product $S_3 \oplus D_4$ of the symmetric group and the dihedral group.
   (a) How many elements does $S_3 \oplus D_4$ have. That is, what is $|S_3 \oplus D_4|$.
   (b) Find the product of $((1,3), H)$ and $((1,2,3), R_{90})$.
   (c) What is the order of the element $((1,3), H)$?
   (d) What is the order of the element $((1,2,3), R_{90})$?

2. Show that $\mathbb{Z}_3 \oplus \mathbb{Z}_5$ is a cyclic group of order 15.
   (Hint: What is the order of the element $(1,1)$?)

3. Is $\mathbb{Z}_2 \oplus \mathbb{Z}_6$ a cyclic group? Explain.

Lecture 22

Symmetry & Counting I: The Orbit-Stabilizer Theorem

In this lecture we discuss how to use group theory to *count like a professional*. The Orbit-Stabilizer Theorem gives an application of cosets to determine the size of a permutation group. In particular, we discover a straightforward way to count the number of symmetries of various geometric objects.

22.1 Orbits & Stabilizers

In this section we will take a look at how permutation groups act on various structures.

It will be helpful to extend the definition of a permutation from finite sets of numbers $[n]$ to arbitrary sets. Let X be a nonempty set. A **permutation** α of X is a bijection $\alpha : X \to X$. The set of all permutations of X is called the **symmetric group of** X and is denoted by S_X:

$$S_X = \{\alpha \mid \alpha : X \to X \text{ is a bijection}\}.$$

If $X = [n] = \{1, 2, \ldots, n\}$ then we simply denoted $S_{[n]}$ by S_n.

Definition 22.1 (Stabilizer of a Point): Let G be a subgroup of S_X. For each $i \in X$, let

$$\text{stab}_G(i) = \{\alpha \in G \mid \alpha(i) = i\}.$$

We call $\text{stab}_G(i)$ the **stabilizer of** i **in** G.

We can check that $\text{stab}_G(i)$ is a subgroup of G by using the Two-Step Subgroup Test (Theorem 11.1). Since ε fixes every element in X it is definitely in $\text{stab}_G(i)$. Let $\alpha, \beta \in G$, then $\alpha(i) = i$ and $\beta(i) = i$. It then follows that $\alpha^{-1}(i) = i$ and $(\alpha\beta)(i) = \beta(\alpha(i)) = \beta(i) = i$, hence $\alpha^{-1}, \alpha\beta \in \text{stab}_G(i)$. Therefore $\text{stab}_G(i) < G$.

Definition 22.2 (Orbit of a Point): Let G be a subgroup of S_X. For each $i \in X$, let

$$\text{orb}_G(i) = \{\alpha(i) \mid \alpha \in G\}.$$

We call $\text{orb}_G(i)$ the **orbit of i under** G.

Example 22.1: If $G = S_4$, then $\text{stab}_{S_4}(3)$ is the set of all permutation in S_4 which fixes 3. There are $4! = 24$ permutations in S_4 but only the ones that don't have 3 in their disjoint cycle form fix 3. Therefore,

$$\begin{aligned}\text{stab}_{S_4}(3) &= \{\varepsilon, (1,2), (1,4), (2,4), (1,2,4), (1,4,2)\}\\ &= S_{\{1,2,4\}}.\end{aligned}$$

Notice we used the notation $S_{\{1,2,4\}}$ to denote the set of all permutations of the set $\{1,2,4\}$.

Example 22.2: Let

$$\begin{aligned}G &= \langle(1,2,3)(4,5,6)(7,8)\rangle\\ &= \{\varepsilon, (1,2,3)(4,5,6)(7,8), (1,3,2)(4,6,5), (7,8), (1,2,3)(4,5,6), (1,3,2)(4,6,5)(7,8)\}.\end{aligned}$$

be a group of permutation on $X = \{1,2,3,4,5,6,7,8\}$. Then

$$\begin{array}{ll}
\text{orb}_G(1) = \{1,2,3\} & \text{stab}_G(1) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(2) = \{2,3,1\} & \text{stab}_G(2) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(3) = \{3,1,2\} & \text{stab}_G(3) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(4) = \{4,5,6\} & \text{stab}_G(4) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(5) = \{5,6,4\} & \text{stab}_G(5) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(6) = \{6,4,5\} & \text{stab}_G(6) = \{\varepsilon, (7,8)\}\\
\text{orb}_G(7) = \{7,8\} & \text{stab}_G(7) = \{\varepsilon, (1,2,3)(4,5,6), (1,3,2)(4,6,5)\}\\
\text{orb}_G(8) = \{8,7\} & \text{stab}_G(8) = \{\varepsilon, (1,2,3)(4,5,6), (1,3,2)(4,6,5)\}
\end{array}$$

In each case notice that $\text{stab}_G(i)$ is a subgroup of G. Also notice that the orbits are either disjoint or equal. Moreover, the distinct orbits form a partition of X:

$$\{1,2,3\}, \{4,5,6\}, \{7,8\}$$

Let G be a group of permutations on X, and define a relation on X by:

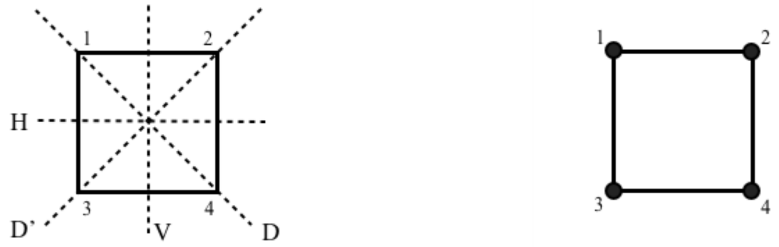
$$x \sim_G y \iff y = \alpha(x) \text{ for some } \alpha \in G. \tag{22.1}$$

Then $\sim_G$ is an equivalence relation (see Exercise 2), and the equivalence class of an element $x \in X$ is its orbit:

$$[x] = \text{orb}_G(x).$$

Since equivalence classes partition the set, this indicates that our observation in Example 22.2 were not a coincidence. Orbits will always be the same or disjoint, and distinct orbit classes will partition X.

Example 22.3: Recall that D_4, the dihedral group of order 8, is the group of all symmetries of the square (see Figure 22.1a). The elements are the rotations $R_0, R_{90}, R_{180}, R_{270}$, and the reflections H, V, D, D'. We can view D_4 as a group of permutations on the vertices of the square. Here we identify the vertices of the square with the set $X = \{1,2,3,4\}$. Since vertex 1 can be taken to any other vertex by a rotation then the orbit of 1 is all of X: $\text{orb}_{D_4}(1) = \{1,2,3,4\}$. See Figure 22.1b.



(a) Reflection elements in D_4

(b) Orbit of vertex 1

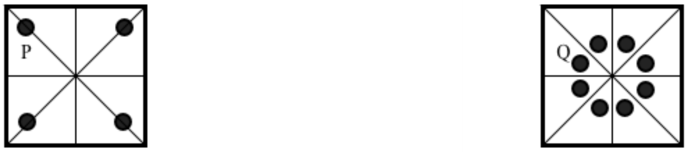
Figure 22.1: The group D_4 acting as a permutation group on the set of vertices.

The stabilizer of 1 is:

$$\text{stab}_{D_4}(1) = \{R_0, D\}.$$

Similarly, we have $\text{stab}_{D_4}(2) = \text{stab}_{D_4}(3) = \{R_0, D'\}$.

Example 22.4: Building on the previous example, we may view D_4 as a group of permutations of the points X enclosed by the square. Figure 22.2a illustrates the orbit of the point P and Figure 22.2b illustrates the orbit of the point Q under D_4. Notice $\text{stab}_{D_4}(P) = \{R_0, D\}$, and $\text{stab}_{D_4}(Q) = \{R_0\}$.



(a) Orbit of point P under action of D_4

(b) Orbit of point Q under action of D_4

Figure 22.2: The group D_4 acting as a permutation group on the set of points enclosed by the square.

We can also view D as a group of permutations on the set of 4 line segments h, v, d, d' shown in Figure 22.3. Then

$$\begin{array}{ll}
\text{orb}_{D_4}(h) = \{h, v\} & \text{stab}_{D_4}(h) = \{R_0, R_{180}, H, V\} \\
\text{orb}_{D_4}(v) = \{h, v\} & \text{stab}_{D_4}(v) = \{R_0, R_{180}, H, V\} \\
\text{orb}_{D_4}(d) = \{d, d'\} & \text{stab}_{D_4}(d) = \{R_0, R_{180}, D, D'\} \\
\text{orb}_{D_4}(d') = \{d, d'\} & \text{stab}_{D_4}(d') = \{R_0, R_{180}, D, D'\}
\end{array}$$

Example 22.5: Let RC_3 be the Rubik's cube group, and let X be the set of all cubies of Rubik's cube. X can be partitioned into edge cubies E, corner cubies V, and centre cubies C. If x denotes the uf edge cubie, then since it is possible to move it to the location of any other edge cubie, then $\text{orb}_{RC_3}(x) = E$. Also, since centre cubies don't move under cube moves, the orbit of each centre cubie is just a set of size 1.

Example 22.6: Again, let RC_3 be the Rubik's cube group, but now let X be the set of all facets of Rubik's cube. Recall $|X| = 48$. The Rubik's cube group can be viewed as a group of permutations of the set X (we have made use of this fact frequently already). Let x be the facet on the *up* layer of the uf cubie. In our numbering system we denoted this facet by $x = 7$. Since an edge cubie can be moved to the location

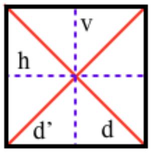


Figure 22.3: Orbit classes of the group D_4 acting as a permutation group on the set of line segments h, v, d, d'.

of any other edge cubie, and with either orientation, then the orbit of x is every edge-facet. Therefore, $|\text{orb}_{RC_3}(7)| = 24$. The next theorem will tell us that $|\text{stab}_{RC_3}(7)| = \frac{|RC_3|}{24}$.

Looking back at the examples we can observe an obvious relationship between the sizes of G, $\text{orb}_G(i)$, and $\text{stab}_G(i)$: we always get $|\text{orb}_G(i)| \cdot |\text{stab}_G(i)|$ equal to the size of G. This is true in general and is stated in the following theorem.

Theorem 22.1 (Orbit-Stabilizer Theorem): *Let G be a subgroup of S_X. Then for any i in X,*

$$|G| = |orb_G(i)| \cdot |stab_G(i)|.$$

Proof: Since $\text{stab}_G(x)$ is a subgroup of G, we know from Lagrange's Theorem that

$$|G|/|\text{stab}_G(x)| = \text{ the number of distinct right cosets of } \text{stab}_G(x) \text{ in } G.$$

So we need to show that the number of right cosets equals the number of elements in $\text{orb}_G(x)$. To this end define

$$\psi : \{(\text{stab}_G(x))\alpha \mid \alpha \in G\} \to \text{orb}_G(x)$$

by

$$\psi(\text{stab}_G(x)\ \alpha) = \alpha(x).$$

Our goal is to show that ψ is a bijection.

(a) ψ **is well defined.** We have

$$\begin{aligned} \text{stab}_G(x)\ \alpha = \text{stab}_G(x)\ \beta \quad &\Longrightarrow \quad \alpha = \gamma\beta \quad \text{for some } \gamma \in \text{stab}_G(x) \\ &\Longrightarrow \quad \alpha(x) = (\gamma\beta)(x) = \beta(\gamma(x)) \\ &\Longrightarrow \quad \alpha(x) = \beta(x) \quad \text{since } \gamma \in \text{stab}_G(x). \end{aligned}$$

(b) ψ **is injective.** Let $\alpha, \beta \in G$, we have

$$\begin{aligned} \psi(\text{stab}_G(x)\ \alpha) = \psi(\text{stab}_G(x)\ \beta) \quad &\Longrightarrow \quad \alpha(x) = \beta(x) \\ &\Longrightarrow \quad \beta^{-1}(\alpha(x)) = x \\ &\Longrightarrow \quad (\alpha\beta^{-1})(x) = x \\ &\Longrightarrow \quad \alpha\beta^{-1} \in \text{stab}_G(x) \\ &\Longrightarrow \quad \text{stab}_G(x)\ \alpha = \text{stab}_G(x)\ \beta. \end{aligned}$$

The last implication follows from Lemma 18.2 (but expressed in terms of right cosets).

(c) **ψ is surjective.** Let $y \in \text{orb}_G(x)$. Then for some $\alpha \in G$ we have $y = \alpha(x)$. Therefore,

$$\psi(\text{stab}_G(x)\ \alpha) = \alpha(x) = y,$$

and so ψ is surjective.

Therefore ψ is a bijection, and so it follows that

$$\begin{aligned} |\text{orb}_G(x)| &= |\{(\text{stab}_G(x))\alpha \mid \alpha \in G\}| \\ &= \text{ the number of right cosets of } \text{stab}_G(x) \text{ in } G \\ &= |G|/|\text{stab}_G(x)|, \end{aligned}$$

which implies

$$|G| = |\text{orb}_G(i)| \cdot |\text{stab}_G(i)|.$$

□

We now consider a few applications of this theorem.

22.2 Permutations Acting on Sets: Application of the Orbit-Stabilizer Theorem

The orbit-stabilizer theorem (Theorem 22.1) is a counting theorem. It enables one to determine the number of elements in a group. We will now see how this theorem will help us determine the number of rotational symmetries of some familiar 3-dimensional objects.

For a object X we let G_X be the group of all rotational symmetries of X. That is, the set of all ways the object can be picked up, rotated, and placed back on a table in front of you, so that it looks as though it wasn't moved. For each of the objects below we will determine $|G_X|$.

22.2.1 Rotation Group of a Tetrahedron

Let G_T be the group of all rotational symmetries of a regular tetrahedron.

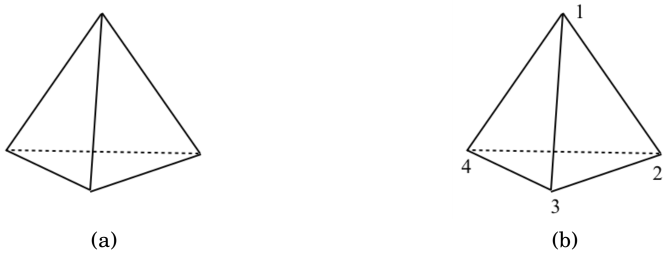
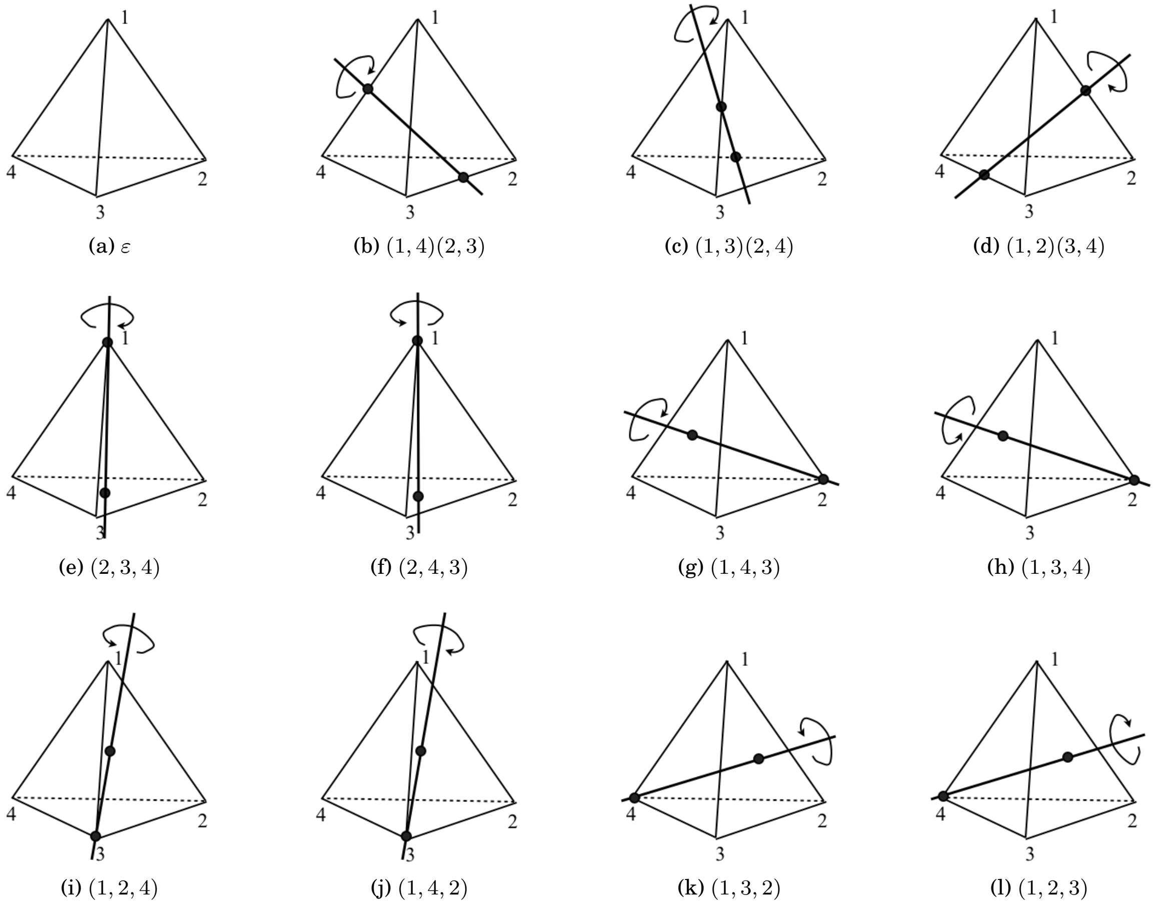


Figure 22.4: regular tetrahedron.

Let V_T be the set of 4 vertices of the tetrahedron, labeled as in Figure 23.4b. Then each rotation in G_T induces a permutation on V_T. That is, each element of G_T gives a permutation in $S_{V_T} = S_4$. Vertex 1 can be taken to any other vertex by a rotation, so the orbit of vertex 1 is $\text{orb}_{G_T}(1) = \{1, 2, 3, 4\}$, and therefore $|\text{orb}_{G_T}(1)| = 4$. The stabilizer of 1 consists satisfies $|\text{stab}_{G_T}(1)| = 3$, and the rotations in the stabilizer are: the identity, and two rotations corresponding to the permutations $(2, 3, 4)$ and $(2, 4, 3)$. Therefore, by the orbit-stabilizer theorem:

$$|G_T| = |\text{orb}_{G_T}(1)| \cdot |\text{stab}_{G_T}(1)| = 4 \cdot 3 = 12.$$

The 12 rotations of G_T are shown in Figure 22.5. Each rotation is described by the permutation it induces on the vertices. It is clear from this description that $G_T \approx A_4$.



(a) ε (b) $(1,4)(2,3)$ (c) $(1,3)(2,4)$ (d) $(1,2)(3,4)$

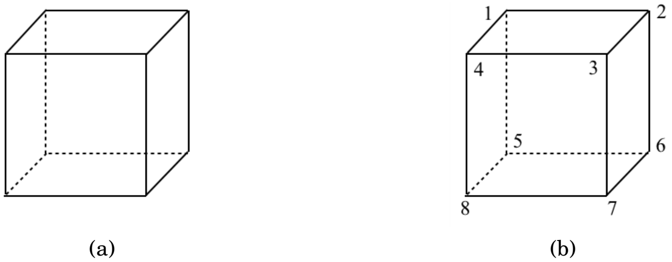
(e) $(2,3,4)$ (f) $(2,4,3)$ (g) $(1,4,3)$ (h) $(1,3,4)$

(i) $(1,2,4)$ (j) $(1,4,2)$ (k) $(1,3,2)$ (l) $(1,2,3)$

Figure 22.5: All 12 rotational symmetries of a regular tetrahedron

22.2.2 Rotation Group of a Cube

Let G_C be the group of all rotational symmetries of a cube.



(a) (b)

Figure 22.6: cube.

We can view G_C as a groups of permutations of the 8 corners, that is, as a subgroup of S_8. Observe that

$$\text{orb}_{G_C}(1) = \{1,2,3,4,5,6,7,8\} \quad \Rightarrow \quad |\text{orb}_{G_C}(1)| = 8$$

and that

$$\text{stab}_{G_C}(1) = \{\varepsilon, (2,4,5)(3,8,6), (2,5,4)(3,6,8)\} \quad \Rightarrow \quad |\text{stab}_{G_C}(1)| = 3.$$

The elements of the stabilizer are the rotations about an axis through vertices 1 and 7.

Therefore, by the orbit stabilizer theorem:

$$|G_C| = |\text{orb}_{G_C}(1)| \cdot |\text{stab}_{G_C}(1)| = 8 \cdot 3 = 24.$$

Recall the symmetric group S_4 has 24 elements. Perhaps G_C is S_4 in disguise. To see if it is we should find 4 things in the cube that G_C permutes. There are 4 diagonals as shown in Figure 22.7, and each rotation of the cube permutes these diagonals. In fact, each rotation of the cube can be described precisely by how these diagonals are permuted. Therefore $G_C \approx S_4$.

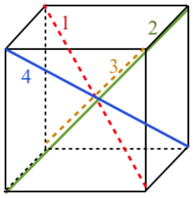
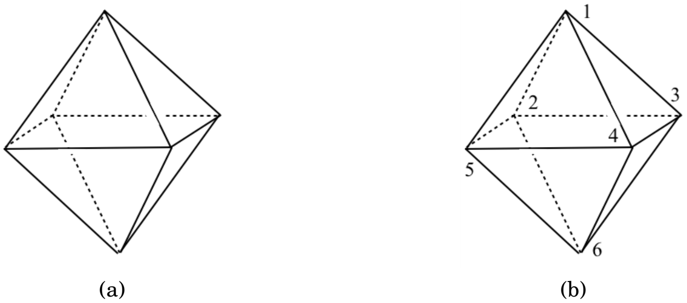


Figure 22.7: Viewing G_C as a group of permutations on the diagonals 1, 2, 3, 4.

22.2.3 Rotation Group of an Octahedron

Let G_O be the group of all rotational symmetries of a regular octahedron.



(a)

(b)

Figure 22.8: regular octahedron.

We can view G_O as a groups of permutations of the 6 vertices, that is as a subgroup of S_6. Observe that

$$\text{orb}_{G_O}(1) = \{1,2,3,4,5,6\} \quad \Rightarrow \quad |\text{orb}_{G_O}(1)| = 6$$

and that

$$\text{stab}_{G_O}(1) = \{\varepsilon, (2,3,4,5), (2,4)(3,5), (2,5,4,3)\} \quad \Rightarrow \quad |\text{stab}_{G_O}(1)| = 4.$$

The elements of the stabilizer are the rotations about an axis through vertices 1 and 6.

Therefore, by the orbit stabilizer theorem:

$$|G_C| = |\text{orb}_{G_O}(1)| \cdot |\text{stab}_{G_O}(1)| = 6 \cdot 4 = 24.$$

It is no coincidence that this is the same size as the group of symmetries of the cube. Figure 22.9 shows the octahedron sitting inside the cube (join midpoints of every two squares by a line). This means that $G_C \approx G_O$. The cube and the octahedron are referred to as *dual solids*.

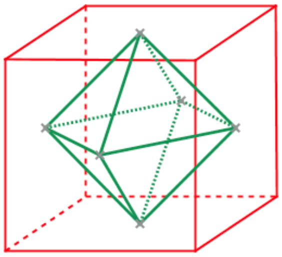


Figure 22.9: The octahedron is dual to the cube, so $G_O \approx G_C$.

22.2.4 Rotation Group of an Dodecahedron

Let G_D be the group of all rotational symmetries of a regular dodecahedron.

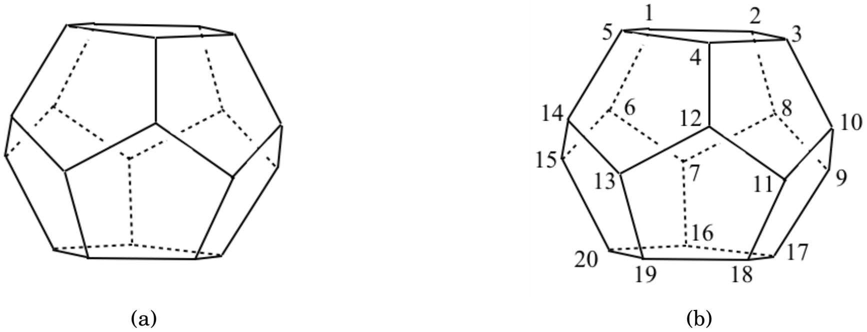


Figure 22.10: regular dodecahedron.

We can view G_D as a groups of permutations of the 20 vertices, that is as a subgroup of S_{20}. Observe that

$$\text{orb}_{G_D}(1) = \{1, 2, 3, \ldots, 20\} \quad \Rightarrow \quad |\text{orb}_{G_D}(1)| = 20$$

and that

$$|\text{stab}_{G_D}(1)| = 3.$$

The elements of the stabilizer are the rotations about an axis through vertices 1 and 18.

Therefore, by the orbit stabilizer theorem:

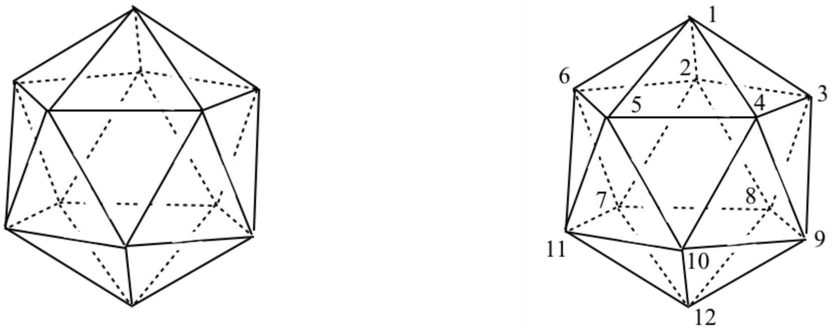
$$|G_C| = |\text{orb}_{G_D}(1)| \cdot |\text{stab}_{G_D}(1)| = 20 \cdot 3 = 60.$$

22.2.5 Rotation Group of an Icosahedron

Let G_I be the group of all rotational symmetries of a regular icosahedron.

We can view G_I as a groups of permutations of the 12 vertices, that is as a subgroup of S_{20}. Observe that

$$\text{orb}_{G_I}(1) = \{1, 2, 3, \ldots, 12\} \quad \Rightarrow \quad |\text{orb}_{G_I}(1)| = 12$$



(a)

(b)

Figure 22.11: regular icosahedron.

and that

$$|\text{stab}_{G_I}(1)| = 5.$$

The elements of the stabilizer are the rotations about an axis through vertices 1 and 12.

Therefore, by the orbit stabilizer theorem:

$$|G_C| = |\text{orb}_{G_I}(1)| \cdot |\text{stab}_{G_I}(1)| = 20 \cdot 3 = 60.$$

It is no coincidence that this is the same size as the group of symmetries of a regular dodecahedron. Figure 22.12 shows both the dodecahedron sitting inside the icosahedron (join midpoints of every two squares by a line), and the icosahedron sitting inside the dodecahedron. This means that $G_I \approx G_D$.

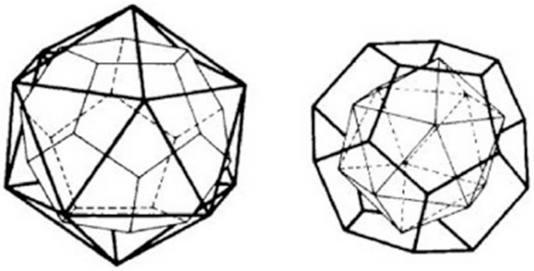


Figure 22.12: The icosahedron is dual to the dodecahedron, so $G_I \approx G_D$.

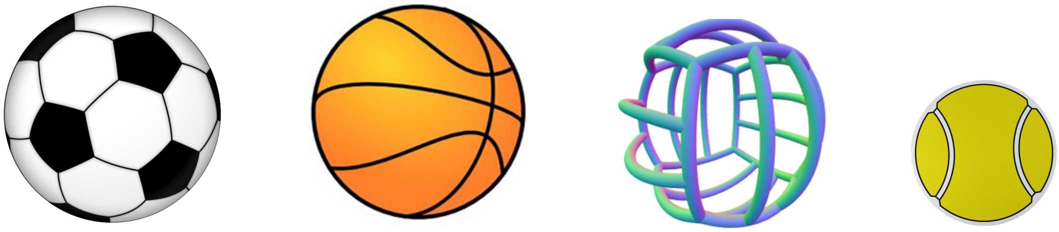
22.2.6 Rotation Group of an Soccer Ball, Basket Ball, Volley Ball, and Tennis Ball

The balls used in soccer, basketball, volleyball, and tennis have distinct patterns on their surface. We can use the orbit-stabilizer theorem to determine the rotational groups of symmetries of these patterns.

For each ball, pick an object on the ball: either a point, or shape. Determine the size of the orbit and stabilizer of the point/shape and verify the results in Table 22.1.

It will help if you have a physical ball in your hands. For the soccer ball, there are 12 pentagons (the black faces), and 20 hexagons. See Figure 22.14 for an unfolded view of the soccer ball.

In case you are interested, the rotational group of the soccer ball is A_5.



(a) soccer ball (b) basket ball (c) volley ball (d) tennis ball

Figure 22.13: Familiar sports balls.

ball	**size of group of rotations**
soccer ball	60
basket ball	4
volley ball	12
tennis ball	4

Table 22.1: The size of the rotational group for various playing balls.

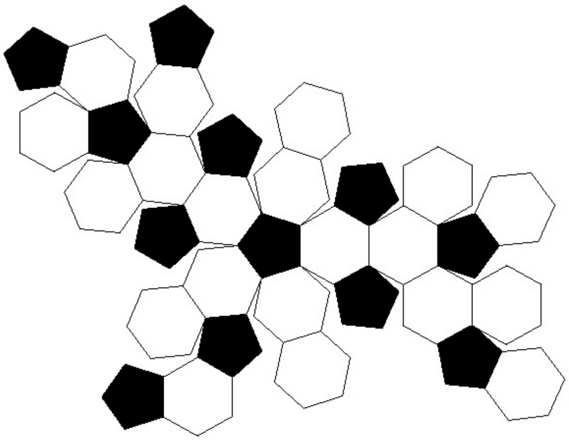
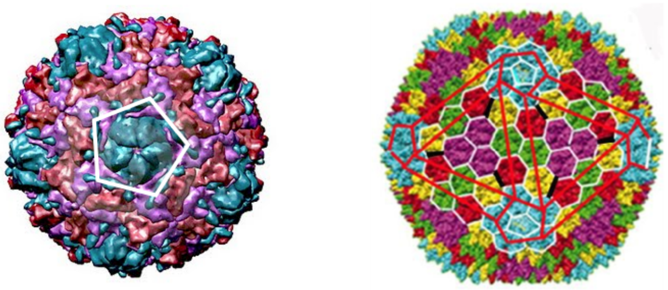


Figure 22.14: A soccer ball unfolded.

In nature, the helix is the structure that occurs most often. The second most commonly found structures are polyhedrons made from pentagons and hexagons, such as the dodecahedron and the truncated icosahedron (soccer ball). Although it is impossible to enclose a space with hexagons along, adding 12 pentagons will be sufficient to enclose the space (like the soccer ball). Many viruses have this kind of structure (Figure 22.15). [1]

[1] John Galloway, *Nature's Second-Favourite Structure.* New Scientist 114 (March 1988); 36-39



(a) rhinovirus (common cold)

(b) Archaeal virus

Figure 22.15: Viruses.

22.3 Exercises

1. Let G be a group of permutations of the set X. For $x \in X$ what type of object is orb(X)? What type of object is stab(x)?

2. Prove the relation defined in (22.1) is an equivalence relation.

3. Let RC_3 be the Rubik's cube group and let H be the subgroup generated by the product $\alpha = UR$.

   $$H = \langle UR \rangle.$$

   Let X be the set of all cubies of the Rubik's cube.

   (a) If x denotes the ufr corner cubie, determine orb$_H(x)$.

   (b) If y denotes the uf edge cubie, determine orb$_H(y)$.

   (c) How many elements do stab$_H(x)$ and stab$_H(y)$ have?

4. Instead of considering the set of vertices of the tetrahedron, consider how G_T permutes the 6 edges of the tetrahedron. By picking one edge, say the edge 12, the edge between vertices 1 and 2, verify that $|\text{orb}_{G_T}(12)| \cdot |\text{stab}_{G_T}(12)| = 12$.

5. Consider how G_T permutes the 4 triangular faces of the tetrahedron. That is, consider G_T as a subgroup of S_4. By picking one face, say the face $f_{1,2,3}$ containing vertices 1, 2 and 3, verify that $|\text{orb}_{G_T}(f_{1,2,3})| \cdot |\text{stab}_{G_T}(f_{1,2,3})| = 12$.

6. Instead of considering the set of vertices of the dodecadedron, consider how G_D permutes the 30 edges of the dodecahedron. That is, consider G_D as a subgroup of S_{30}. By picking one edge, say the edge 12, the edge between vertices 1 and 2, verify that $|\text{orb}_{G_D}(12)| \cdot |\text{stab}_{G_D}(12)| = 60$.

7. Consider how G_D permutes the 12 pentagonal faces of the dodecahedron. That is, consider G_D as a subgroup of S_{12}. By picking one face, say the face f containing vertices $1, 2, 3, 4, 5$, verify that $|\text{orb}_{G_D}(f)| \cdot |\text{stab}_{G_D}(f)| = 60$.

8. For each of the following objects, describe each element of the group of rotations as a single rotation. (Similar to what was done for the tetrahedron in Figure 22.5.)

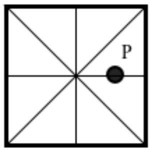
   (a) cube

   (b) octahedron

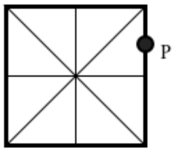
9. Let G be the group of rotations of a rectangular box of dimensions $1 \times 2 \times 3$. Describe each element of G as a rotation.

10. Let G be the group of rotations of a rectangular box of dimensions $1 \times 1 \times 2$. Describe each element of G as a rotation.

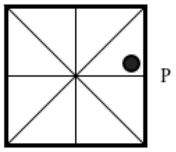
11. The group D_4 acts as a group of permutations of the points enclosed by the square shown below. (The axis of symmetry are drawn for reference purposes.) For each square, locate the points in the orbit of the indicated point P under the action of D_4. In each case, determine the stabilizer of P.



(a)



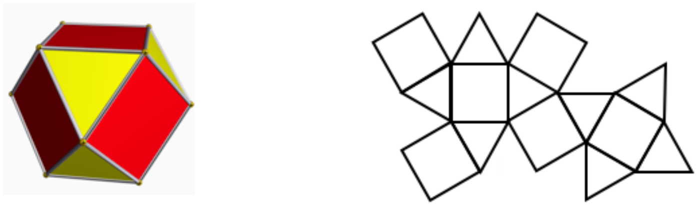
(b)



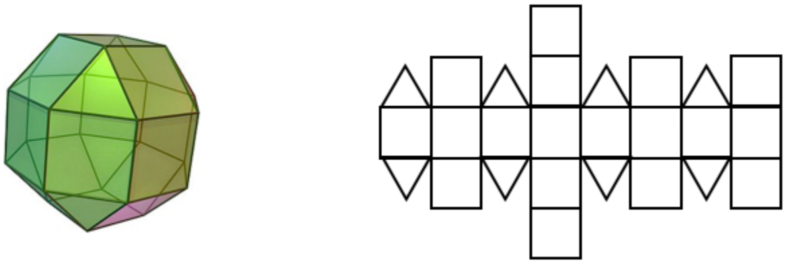
(c)

12. A soccer ball has 20 faces that are regular hexagons and 12 faces that are regular pentagons (see Figures 22.13a and 22.14). Use the orbit stabilizer theorem to explain why a soccer ball cannot have $60°$ rotational symmetry about a line through the centres of two opposite hexagonal faces.

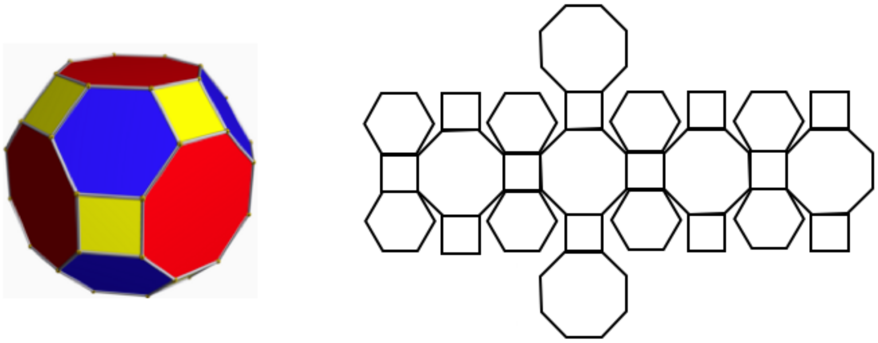
13. For each of the solids below, determine the number of rotational symmetries. (In the figures each solid is also shown as "unfolded".)



(a) cuboctahedron



(b) (small) rhombicuboctahedron



(c) great rhombicuboctahedron or truncated cuboctahedron

Lecture 23

Symmetry & Counting II: Burnside's Theorem

In this lecture we continue our discussion of how to use group theory to *count like a professional*. We look at an application permutation groups to count the number of different designs there are of various objects.

23.1 A Motivating Example

Consider the task of colouring the six vertices of a regular hexagon so that there are three black and three white vertices. Figure 23.1 shows an example of one such colouring.

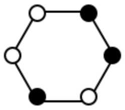
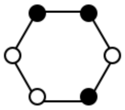
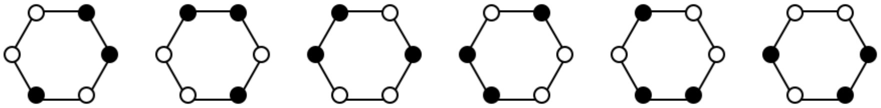


Figure 23.1: An example of a colouring of the vertices of the hexagon: three white, three black.

Ceramic Tiles:
If such a colouring appears on a ceramic tile, it wouldn't make sense to consider this different from the colouring



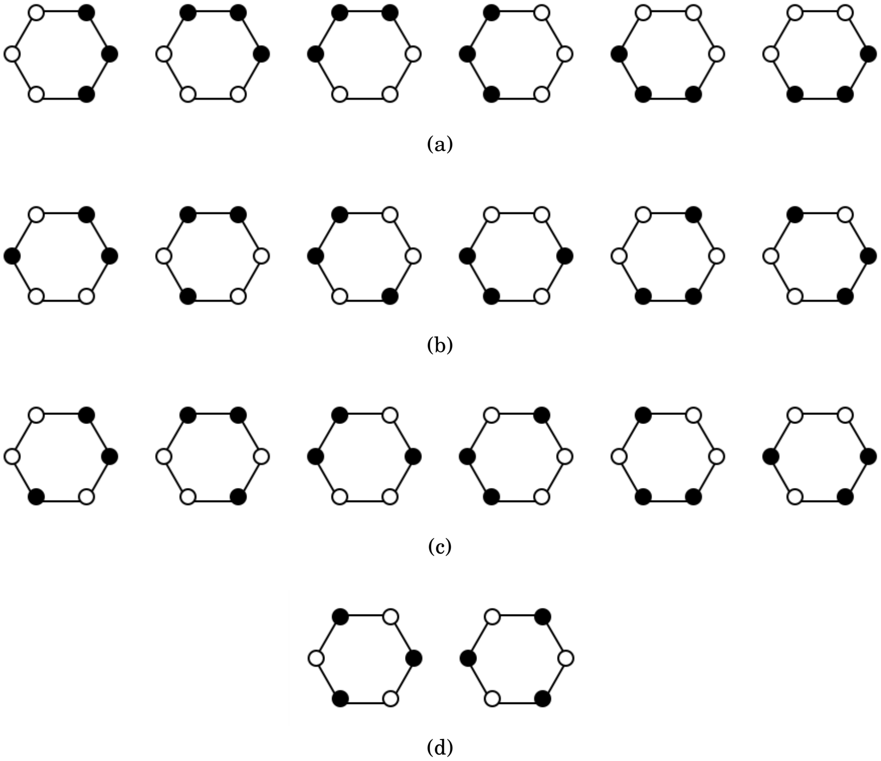
since this one can be obtained by rotating the one in Figure 23.1 counterclockwise by 60°. In this case, we should consider two colours equivalent if one can be obtained from the other by a rotation of the hexagon. In other words, a manufacturer would only need to make the tile in Figure 23.1, and simply by rotating the tile the following six colourings are equivalent under the group of rotations of a hexagon.



How many tiles would a manufacturer need to make in order to obtain all possible ways to colour three vertices black and three white (up to rotational equivalence)?

There are $\binom{6}{3} = 20$ ways to pick three vertices to colour black. As we observed above it would be nonsensical for a manufacturer to produce each of the 20 designs, since up to rotation, the colouring in Figure 23.1 is equivalent to six different designs.

Figure 23.2 shows all 20 possible colourings. They are organized into equivalence classes. For example, all colouring in 23.2a are equivalent under the rotational group of the hexagon. Similarly for the other three cases. This means, a manufacturer would only need to produce 4 different tiles, say for example the first one in each collection of Figure 23.2.



(a)

(b)

(c)

(d)

Figure 23.2: All the different ways to colour three vertices of a hexagon black, and the other three white.

Beads on a Necklace:
On the other hand, if we think of these colourings as representing beads on a necklace, then two colourings would be equivalent if one can be obtained from the other by an element of the dihedral group of the hexagon. In other words, colourings can also be reflected to obtain equivalent colourings. In this situation, the colourings in Figure 23.2b and 23.2c are equivalent. This means there are essentially 3 different ways to make a necklace with three black beads and three white ones (up to rotational/reflexive symmetries of the hexagon).

In general, we say that two designs (arrangements) A and B are *equivalent under a group* G of permutations if there is an element $\alpha \in G$ such that $\alpha(A) = B$. That is, two designs are equivalent under G if they are in the same orbit of G. The set being permuted by G is the set of designs or arrangements.

Therefore the number of inequivalent configurations is the number of orbit classes under G. In the next section we present a celebrated theorem which allows us to count the number of orbit classes.

In the ceramic tile example, the 20 designs in Figure 23.2 have been split up into 4 orbit classes ((a)-(d)) under the group of rotations $\mathbb{Z}_6$ of the hexagon. In the necklace example, there are only 3 orbit classes under the dihedral group D_6. Classes (b) and (c) merge to form one equivalence class in this case.

23.2 Burnside's Theorem

Let X be a nonempty set, and S_X the set of all permutations of X:

$$S_X = \{\alpha \mid \alpha : X \to X \text{ is a bijection}\}.$$

We first recall what we mean by the *fixed set* of a permutation in S_X.

For a permutation $\alpha \in S_X$, the **fixed set of** α is the set of all elements in X that α doesn't move. We denote the set by fix(α).

$$\text{fix}(\alpha) = \{x \in X \mid \alpha(x) = x\}.$$

Note that

$$x \in \text{fix}(\alpha) \Longleftrightarrow \alpha \in \text{stab}_{S_X}(x).$$

Theorem 23.1 (Burnside's Theorem): [a] *If G is a finite group of permutations on a set X, then the number of distinct orbits of G on X is*

$$N = \frac{1}{|G|} \sum_{\alpha \in G} |\textit{fix}(\alpha)|.$$

[a] This theorem is also commonly called the *Polya-Burnside Counting Theorem*.

Proof: Let the orbits be

$$\begin{aligned} \mathcal{O}_1 &= \{a_1, \ldots, a_m\} \\ \mathcal{O}_2 &= \{b_1, \ldots, b_n\} \\ &\vdots \\ \mathcal{O}_N &= \ldots \end{aligned}$$

Recall, the $\mathcal{O}_i$'s partition X, so each element of X appears in one and only one orbit. For each $x \in X$ we apply the orbit-stabilizer theorem to get $|\text{orb}_G(x)| \cdot |\text{stab}_G(x)| = |G|$, or equivalently $|\text{stab}_G(x)| = \frac{|G|}{|\mathcal{O}|}$, where $\mathcal{O} = \text{orb}_G(x)$. Therefore,

$$\mathcal{O}_1 : \qquad |\text{stab}_G(a_1)| + |\text{stab}_G(a_2)| + \cdots + |\text{stab}_G(a_m)| = \underbrace{\frac{|G|}{m} + \cdots + \frac{|G|}{m}}_{m \text{ terms}} = |G|$$

$$\mathcal{O}_2 : \qquad |\text{stab}_G(b_1)| + |\text{stab}_G(b_2)| + \cdots + |\text{stab}_G(b_n)| = \underbrace{\frac{|G|}{n} + \cdots + \frac{|G|}{n}}_{n \text{ terms}} = |G|$$

$$\vdots$$

Summing all these equations, we obtain

$$\sum_{x \in X} |\mathrm{stab}_G(x)| = |G| \cdot N.$$

On the other hand,

$$x \in \mathrm{fix}(\alpha) \Longleftrightarrow \alpha \in \mathrm{stab}_G(x),$$

so

$$\begin{aligned}\sum_{x \in X} |\mathrm{stab}_G(x)| &= |\{(\alpha, x) \mid \alpha \in G, x \in X, \alpha(x) = x\}| \\ &= \sum_{\alpha \in G} |\mathrm{fix}(\alpha)|.\end{aligned}$$

Therefore,

$$\sum_{\alpha \in G} |\mathrm{fix}(\alpha)| = |G| \cdot N$$

so that

$$N = \frac{1}{|G|} \sum_{\alpha \in G} |\mathrm{fix}(\alpha)|.$$

□

23.3 Applications of Burnside's Theorem

Example 23.1: Let's return to the ceramic tile and necklace problems from Section 23.1 and see how to apply Burnside's theorem in this familiar context. It will be convenient to recall that the dihedral group D_6 consists of elements:

$$D_6 = \{\varepsilon, r, r^2, r^3, r^4, r^5, f, rf, r^2f, r^3f, r^4f, r^5f\}$$

where r denotes a clockwise rotation through 60° and f is a reflection about a line through opposite vertices. The groups of rotational symmetries is

$$G = \langle r \rangle = \{\varepsilon, r, r^2, r^3, r^4, r^5\}.$$

In the case of counting hexagonal tiles with three black vertices and three white vertices, the set of objects being permuted is the 20 possible designs, whereas the group of permutations is G, the group of six rotational symmetries of a hexagon.

The identity fixes all 20 designs in Figure 23.2. Rotations through 60°, 180°, or 300° fix none of the designs. That is, $|\mathrm{fix}(r)| = |\mathrm{fix}(r^3)| = |\mathrm{fix}(r^5)| = 0$. Rotations through 120° and 240° fix the two designs in Figure 23.2d, so $|\mathrm{fix}(r^2)| = |\mathrm{fix}(r^4)| = 2$. We summarize these results in Table 23.1.

By Burnside's Theorem, we have that

$$\begin{aligned}\text{number of orbits } = N &= \frac{1}{|G|} \sum_{\alpha \in G} |\mathit{fix}(\alpha)| \\ &= \frac{1}{6}(20 + 0 + 2 + 0 + 2 + 0) \\ &= \frac{24}{6} = 4.\end{aligned}$$

Now let's use Burnside's Theorem to count the number of necklace arrangements. In this case we want to count the number of orbits under D_6. Table 23.2 summarizes the sizes of the fixed sets for each $\alpha \in D_6$.

element: α	**Number of arrangements fixed by this type of element:** $\|\mathit{fix}(\alpha)\|$
ε	20
r	0
r^2	2
r^3	0
r^4	2
r^5	0

Table 23.1: $|\text{fix}(\alpha)|$ for each $\alpha \in \langle r \rangle < D_6$.

type of element	**number of elements of this type**	**Number of arrangements fixed by this type of element**
identity	1	20
rotation of order 2 (180°)	1	0
rotation of order 3 (120° or 240°)	2	2
rotation of order 6 (60° or 300°)	2	0
reflection across diagonal	3	4
reflection across bisector	3	0

Table 23.2: $|\text{fix}(\alpha)|$ for each type of $\alpha \in D_6$.

By Burnside's Theorem, we have that

$$\begin{aligned}\text{number of orbits } = N &= \frac{1}{|D_6|}\sum_{\alpha \in D_6} |\mathit{fix}(\alpha)| \\ &= \frac{1}{12}(20 + 0 + 2\cdot 2 + 0 + 3\cdot 4 + 0) \\ &= \frac{36}{12} = 3.\end{aligned}$$

Example 23.2: Consider the number of ways to colour the faces of a regular tetrahedron with 4 different colours.

How should we decide when two colourings of the tetrahedron are nonequivalent? Certainly, if we were to pick up a tetrahedron coloured in a certain manner, rotate it, and put it back down, we would think of the tetrahedron as being positioned differently rather than being coloured differently. So our permutation group for this problem is just the group of 12 rotations of the tetrahedron, which we denoted by G_T (see Section 22.2.1). This group consists of the identity; eight elements of order 3, each which fix one vertex; and three elements of order 2, each which fix no vertices (but fix exactly two edges).

The total number of colourings, without regard to equivalence, is $4!$. Therefore

$$\text{fix}(\varepsilon) = 4!$$

while, for any $\alpha \in G_T$, $\alpha \neq \varepsilon$,

$$\text{fix}(\alpha) = 0.$$

Table 23.3 summarizes the results.

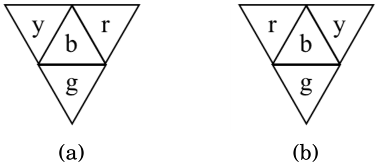
type of element	number of elements of this type	Number of arrangements fixed by this type of element
identity	1	4!
rotation of order 2	3	0
rotation of order 3	8	0

Table 23.3: $|\text{fix}(\alpha)|$ for various types of $\alpha \in G_T$.

By Burnside's Theorem, we have that

$$\begin{aligned}\text{number of orbits } &= N = \frac{1}{|G_T|}\sum_{\alpha\in G_T}|\mathit{fix}(\alpha)| \\ &= \frac{1}{12}(4! + 0 + 0 + \cdots 0) \\ &= \frac{4!}{12} = 2.\end{aligned}$$

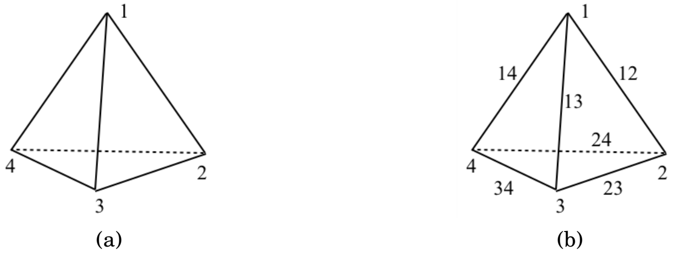
Representative for the two orbit classes are showing in Figure 23.3.



(a) (b)

Figure 23.3: The two inequivalent colourings of the faces of a tetrahedron (tetrahedron is unfolded to see all sides).

Example 23.3: Suppose that we have the colours red (R), green (G), and blue (B), and we wish to colour the edges of a regular tetrahedron. First observe there are $3^6 = 729$ colourings without regard to equivalence. As with the previous example, we consider how the group of rotations of the tetrahedron, G_T acts on these colourings. Two colourings are equivalent if they are in the same G_T orbit. Every rotation permutes the 729 colourings, and to apply Burnside's theorem we must determine the size of fix(α) for each of the 12 rotations.



(a) (b)

Figure 23.4: regular tetrahedron, with vertices and edges labeled.

The identity fixes all 729 colourings:

$$|\text{fix}(\varepsilon)| = 729.$$

Now consider the rotation $(2, 3, 4)$ or order 3. (Here we are describing a rotation by the permutation it induces on the vertices.) Suppose that a specific colouring is fixed by this element (that is, the tetrahedron appears to be coloured the same before and after this rotation). Since $(2,3,4)$ takes edge 12 to edge 13, edge 13 to 14, and edge 14 to edge 12, these three edges must be coloured the same. The same argument shows that edges 23, 24 and 34 must be coloured the same. Since there are three choices of colour for each of these two sets, there are $3^2 = 9$ colourings of the tetrahedron in total that are fixed by the rotation $(2,3,4)$. Table 23.4 lists the 9 different colourings. Therefore,

$$|\text{fix}((2,3,4))| = 9.$$

For each of the other 7 rotations of order 3 a similar argument shows the fixed set has sized 9.

	egde colours					
colouring	12	13	14	23	24	34
scheme 1	R	R	R	R	R	R
scheme 2	R	R	R	G	G	G
scheme 3	R	R	R	B	B	B
scheme 4	G	G	G	G	G	G
scheme 5	G	G	G	R	R	R
scheme 6	G	G	G	B	B	B
scheme 7	B	B	B	B	B	B
scheme 8	B	B	B	R	R	R
scheme 9	B	B	B	G	G	G

Table 23.4: Nine colourings fixed by $(2,3,4)$.

Now consider the rotation $(1,2)(3,4)$ of order 2. Since edges 12 and 34 are fixed they may be coloured in any way and will appear the same after the rotation $(1,2)(3,4)$ (since the rotation fixes these edges). This gives $3 \cdot 3 = 9$ choices for these edges. Edges 14 and 23 are swapped by the rotation $(1,2)(3,4)$ and so must be coloured the same. Similarly, edges 13 and 24 are swapped and must be coloured the same. There are 3 choices to colour each of these sets, so there are 9 ways to colour these two sets altogether. Therefore, there are $9 \cdot 9 = 81$ ways to colour all the edges in such a way that the colouring remains fixed under the rotation $(1,2)(3,4)$. Table 23.5 lists the 81 different colourings. Therefore,

$$|\text{fix}((1,2)(3,4))| = 81.$$

For each of the other 2 rotations of order 2 a similar argument shows the fixed set has sized 81.

	egde colours					
colouring	12	13	14	23	24	34
scheme 1	X	Y	R	R	R	R
scheme 2	X	Y	R	R	G	G
scheme 3	X	Y	R	R	B	B
scheme 4	X	Y	G	G	G	G
scheme 5	X	Y	G	G	R	R
scheme 6	X	Y	G	G	B	B
scheme 7	X	Y	B	B	B	B
scheme 8	X	Y	B	B	R	R
scheme 9	X	Y	B	B	G	G

Table 23.5: Eighty-one colourings fixed by $(1,2)(3,4)$. X and Y can be any of R, G, B.

The results are summarized in Table 23.6.

type of element	number of elements of this type	Number of arrangements fixed by this type of element
identity	1	729
rotation of order 2	3	81
rotation of order 3	8	9

Table 23.6: $|\text{fix}(\alpha)|$ for various types of $\alpha \in G_T$.

By Burnside's Theorem, we have that

$$\begin{aligned}\text{number of orbits } = N &= \frac{1}{|G_T|}\sum_{\alpha \in G_T} |\mathit{fix}(\alpha)| \\ &= \frac{1}{12}(729 + 3(81) + 8(9)) \\ &= \frac{1044}{12} = 87.\end{aligned}$$

It would be a difficult task to solve this problem without Burnside's Theorem.

At this point you may be wondering who besides mathematicians would be interested in counting problems such as these. Chemists for one, are interested in these types of counting problems. Though, their interests lie more in counting configurations of molecules. We'll now look at an example.

Example 23.4: Benzene is a chemical compound, each molecule of shich is made up of six carbon (C) atoms, and six hydrogen (H) atoms. The carbon atoms are arranged in a hexagon with alternating single and double bonds. Each carbon atom must have four bonds and each hydrogen atom must have one bond. See Figure 23.5.

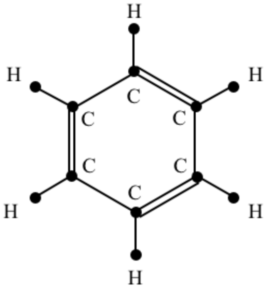


Figure 23.5: A benzene molecule.

By replacing three of the hydrogen atoms by CH_3 clusters (see Figure 23.6) we can create a chemical derivative from benzene. Let's determine the number of such derivatives.

Taking into account orientation, the number of possibilities would just be the number of ways of choosing three hydrogen atoms for replacement from six possibilities, in other words $\binom{6}{3} = 20$. However, those that are related by rotational/reflective symmetry clearly correspond to the same derivative chemical. So we wish to determine the number of derivatives up to equivalence under the symmetries of the molecule.

Let r denote the rotation in the clockwise direction through an angle of $120°$, and let f be a reflection about the "$x - axis$". The group of symmetries of the molecule (respecting the single/double bonds) is:

$$G = \{\varepsilon, r, r^2, f, rf, r^2f\}.$$

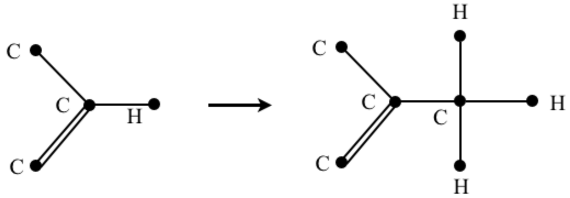


Figure 23.6: An example of how a hydrogen atom is replaced with a CH_3 cluster.

A reflection (f, rf, or r^2f) swaps pairs of vertices so it would not fix any molecule. An order 3 rotation would fix a molecule if all clusters lie on the same side of a double bond. There are two such molecules. Table 23.7 summarizes the number of arrangements, where three hydrogen atoms are replaced by CH_4 clusters, which are fixed by each element of G.

type of element	number of elements of this type	Number of arrangements fixed by this type of element
identity	1	20
reflection of order 2: f, rf, r^2f,	3	0
rotation of order 3: r, r^2	2	2

Table 23.7: $|\text{fix}(\alpha)|$ for various types of $\alpha \in G$.

By Burnside's Theorem, we have that

$$\begin{aligned}\text{number of orbits } = N &= \frac{1}{|G|}\sum_{\alpha\in G}|\mathit{fix}(\alpha)| \\ &= \frac{1}{6}(20+3(0)+2(2)) \\ &= \frac{24}{6} = 4.\end{aligned}$$

Therefore, there are 4 such derivatives. You should try listing them.

Another kind of molecule that chemists consider is visualized as a regular tetrahedron with a carbon atom at the centre and any of the four radicals $HOCH_2$ (hydroxymethyl), C_2H_2 (ethyl), Cl (chlorine) or H (hydrogen) at the four vertices. The number of such molecules can be easily counted using Burnside's Theorem.

23.4 Exercises

1. Define the functions fix, mov, stab and orb.
2. Determine the number of different ways there are of arranging 6 keys on a key ring.
3. Determine the number of ways of colouring the vertices of a square so that two are red and two are green.
4. Determine the number of ways there are to colour the vertices of a pentagon under each of the following conditions:

(a) using five distinct colours;

(b) using colours black and white, so that two are black and three are white;

(c) using colours black, white and blue, so that two are black, two are white, and one is blue.

5. Determine the number of ways of colouring a regular n-gon with n different colours.

6. Determine the number of ways of seating n diplomats around a table.

7. Determine the number of (inequivalent) ways to colour the 6 faces of a cube with 6 distinct colours. Consider two colourings equivalent if one can be obtained from the other by a rotation of the cube.

8. Determine the number of (inequivalent) ways to colour the 6 faces of the cube so that **three** faces are white and **three** faces are black. Consider two colourings equivalent if one can be obtained from the other by a rotation of the cube.

9. Determine the number of (inequivalent) ways to colour the 6 faces of the cube so that **two** faces are white and **four** faces are black. Consider two colourings equivalent if one can be obtained from the other by a rotation of the cube.

10. Determine the number of (inequivalent) ways to colour the 12 edges of the cube so that **six** edges are white and **six** edges are black. Consider two colourings equivalent if one can be obtained from the other by a rotation of the cube.

11. Determine the number of (inequivalent) ways to colour the 12 pentagonal faces of a regular dodecahedron with 12 distinct colours. Consider two colourings equivalent if one can be obtained from the other by a rotation of the dodecahedron.

12. Determine the number of (inequivalent) ways to colour the 12 pentagonal faces of a regular dodecahedron so that 6 faces are white and 6 faces are black. Consider two colourings equivalent if one can be obtained from the other by a rotation of the dodecahedron.

13. Determine the number of (inequivalent) ways to colour the 20 triangular faces of a regular icosahedron with 20 different colours. Consider two colourings equivalent if one can be obtained from the other by a rotation of the icosahedron.

14. A benzene molecule can be viewed as six carbon atoms arranged in a regular hexagon. See Figure 23.7 (ignore double vs. single bonds). At each carbon atom, one of three radicals (NH_2, $COOH$, or OH) can be attached. How many such compounds are possible?

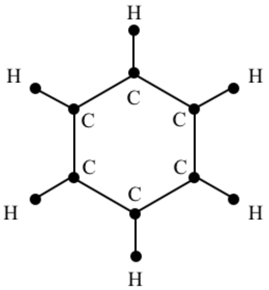


Figure 23.7: Diagram for Exercise 14.

Lecture 24

Lights Out

In this lecture we look at an electronic puzzle called *Lights Out* and see how we can solve it using linear algebra.

24.1 Lights Out

Lights Out consists of a 5-by-5 grid of lights; when the game starts, a set of these lights (random, or one of a set of stored puzzle patterns) are switched on. Pressing one of the lights will toggle it, and the four lights adjacent to it, on and off. (Diagonal neighbours are not affected.) The game provides a puzzle: given some initial configuration where some lights are on and some are off, the goal is to switch all the **lights off**, preferably in as few button presses as possible. See Figure 24.1 for sample game play.

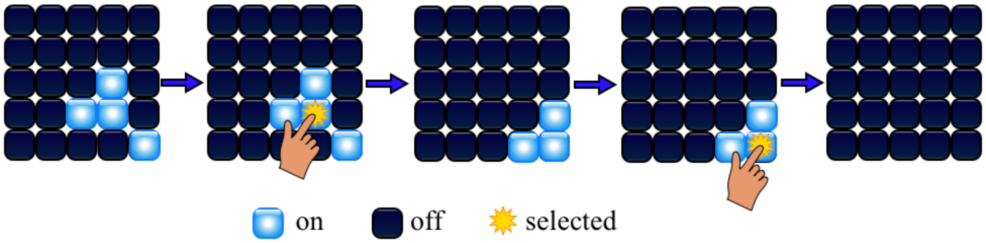


Figure 24.1: A demonstration of Lights Out play.

Two physical versions of the game are shown in Figure 24.2. The first one is the original game, each button has two states: on or off. The second one, called Lights Out 2000, has a further option of allowing 3 states for each button: red, green and off.

Variations of Lights Out: Lights Out is another puzzle that has been updated for the digital era. Many variations of this puzzle exist now in software form. Variations include: more states for the lights (i.e. more colours for the lights to cycle through), changing size of game boards, modifying how a button press changes the state of the lights. For example, we could make it so pressing a button changes the state of all lights in the same row and column as the button that was pressed.

Software: This puzzle is available for play on the web as well as available for ios. More information about where to find your own digital copy can be found here http://www.sfu.ca/ jtmulhol/math302/.

(a) Lights Out

(b) Lights Out 2000

Figure 24.2: Lights Out electronic games released by Tiger Toys

24.2 Lights Out: A Matrix Model

A complete strategy for the game can be obtained using linear algebra, requiring only knowledge of Gauss-Jordan elimination and some facts about the column and null space of a matrix.

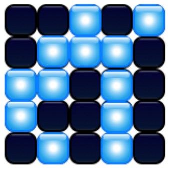
We make some initial observations:

(a) Pushing a button twice is equivalent to not pushing it at all.

(b) The state of a button depends only on how often (whether even or odd) it and its neighbours have been pushed. Hence, the order in which the buttons are pressed does not matter. Together with (a), for any configuration, a solution exists in which each button is pushed no more than once.

We will represent the state of each light by an element of $\mathbb{F}_2 = \{0, 1\}$; 1 for on, 0 for off. We can represent a lit button configuration by a 5×5 matrix A with entries from $\mathbb{F}_2$, i.e. $A \in M_{5\times5}(\mathbb{F}_2)$ where the $(i, j)^{\text{th}}$ entry is 1 if the button in position (i, j) is on, or 0 if the button is off. See Figure 24.3. We call this matrix the lit button **configuration matrix**. Here,

$$M_{5\times5}(\mathbb{F}_2) = \{[b_{i,j}] \mid 1 \le i, j \le 5, b_{i,j} \in \mathbb{F}_2 = \{0, 1\}\}.$$



(a) sample lit button configuration

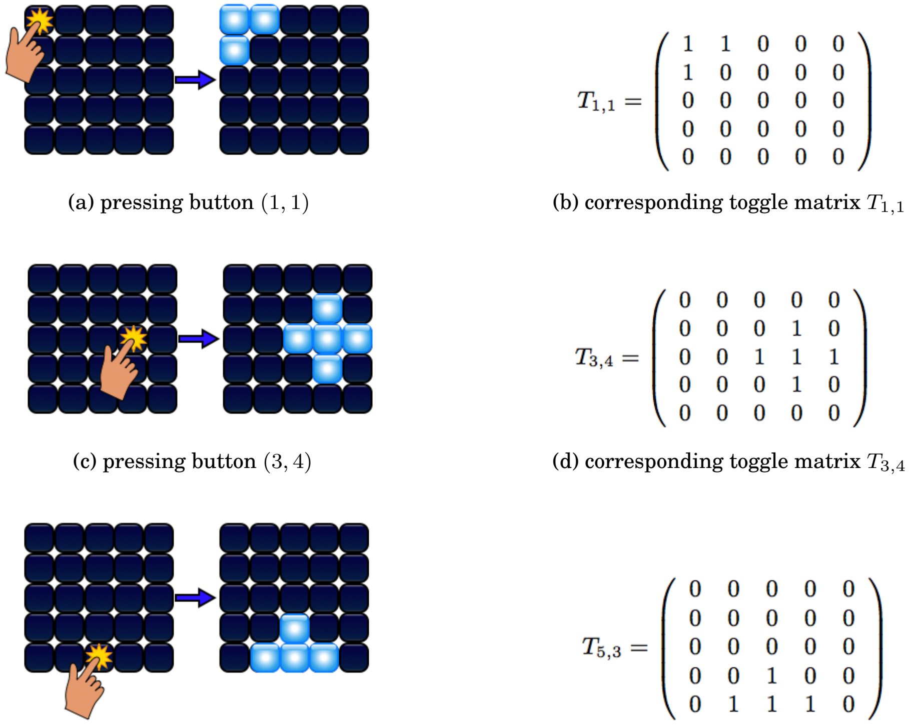
$$\begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

(b) corresponding configuration matrix in $M_{5\times5}$

Figure 24.3: Matrix corresponding to a lit button configuration

If a button is pressed the states of the lights around the button are toggled. For the standard lights out puzzle it is the button itself, and its vertical and horizontal neighbours that are toggled. For each button (i, j) we define a **toggle matrix** $T_{i,j}$ where the entry is 1 if the button in that location changes state, or 0 if it doesn't. For example, see Figure 24.4.

The sample game play shown in Figure 24.1 can be translated into a matrix equation using configuration and toggle matrices as follows.



(a) pressing button $(1,1)$

$$T_{1,1} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(b) corresponding toggle matrix $T_{1,1}$

(c) pressing button $(3,4)$

$$T_{3,4} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(d) corresponding toggle matrix $T_{3,4}$

(e) pressing button $(5,3)$

$$T_{5,3} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

(f) corresponding toggle matrix $T_{5,3}$

Figure 24.4: Some examples of the toggle matrix corresponding to pressing a button.

Let B be the initial configuration matrix:

$$B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then the game play corresponds to

$$B + T_{4,4} + T_{5,5} = \mathbf{0},$$

where $\mathbf{0}$ denotes the zero matrix, and addition of matrices is done in the usual way – componentwise – but here entries are added modulo 2. Recall, modulo 2 arithmetic means $0+0=0$, $0+1=1+0=1$, $1+1=0$. Since a matrix in $M_{5\times 5}(\mathbb{F}_2)$ added to itself is $\mathbf{0}$ (Why?) then adding B to both sides of the previous equation gives

$$T_{4,4} + T_{5,5} = B.$$

In other words, to solve the puzzle we just have to determine how to write B as a linear combination of the toggle matrices.

Moreover, since for any matrices $A, C \in M_{5\times 5}(\mathbb{F}_2)$ we have $A + C = C + A$ and $A + A = \mathbf{0}$ then we can now easily see why (i) order in which buttons are presses doesn't matter, and (ii) no button needs to be pressed more than once.

In general, given any lit button configuration $B = [b_{i,j}]$, solving the puzzle is equivalent to solving the matrix equation:

$$\sum_{\substack{1 \le i \le 5 \\ 1 \le j \le 5}} x_{i,j} T_{i,j} = B \tag{24.1}$$

for the 25 coefficients $x_{i,j} \in \{0,1\}$. The coefficients $x_{i,j}$ tell use exactly what buttons we need to press. We call $X = [x_{i,j}]$ the **strategy matrix**. We will sometimes write it as a vector $\boldsymbol{x} = (x_{1,1}, x_{1,2}, x_{1,3}, \ldots, x_{4,5}, x_{5,5})$ and call it the **strategy vector**. In general, we can turn any matrix into a vector by listing the entries in order from left-to-right, then top to bottom.

Matrix equation (24.1) corresponds to a system of $5 \cdot 5 = 25$ linear equations (one for each component of the matrix equation).

For example, the linear equation corresponding to entry $(1,1)$ in matrix equation (24.1) is

$$x_{1,1} + x_{1,2} + x_{2,1} = b_{1,1},$$

since the only toggle matrices with 1 in position $(1,1)$ are $T_{1,1}, T_{1,2}$, and $T_{2,1}$. Similarly, the linear equation corresponding to entry $(3,4)$ in matrix equation (24.1) is

$$x_{2,4} + x_{3,3} + x_{3,4} + x_{3,5} + x_{4,4} = b_{3,4}.$$

Writing $\boldsymbol{b} = (b_{1,1}, b_{1,2}, b_{1,3}, \ldots, b_{4,5}, b_{5,5})$ for the vector corresponding to the configuration matrix B, it is straightforward to check that this big system (Equation (24.1)) can be written as a matrix product

$$A\boldsymbol{x} = \boldsymbol{b} \tag{24.2}$$

where A is the 25×25 matrix whose columns are the toggle vectors $\boldsymbol{t}_{i,j}$ (which are the toggle matrices $T_{i,j}$ written as vectors):

$$A = [\boldsymbol{t}_{1,1} \mid \boldsymbol{t}_{1,2} \mid \cdots \mid \boldsymbol{t}_{5,5}].$$

We can write A as

$$A = \begin{pmatrix} C & I_5 & 0 & 0 & 0 \\ I_5 & C & I_5 & 0 & 0 \\ 0 & I_5 & C & I_5 & 0 \\ 0 & 0 & I_5 & C & I_5 \\ 0 & 0 & 0 & I_5 & C \end{pmatrix} \qquad \text{(lights out matrix)} \tag{24.3}$$

where C represents the 5×5 matrix

$$C = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix},$$

and I_5 denotes the 5×5 identity matrix. The matrix A is referred to as the **lights out matrix**.

Therefore, solving the puzzle for a general configuration $\boldsymbol{b}$ is equivalent to solving the 25×25 linear system 24.2 for a strategy vector $\boldsymbol{x}$ (where all arithmetic is done modulo 2).

We would like to determine the answers to the following questions.

(a) Will the standard algorithm using Gauss-Jordan elimination work to solve this system? Recall, this method works if entries are real numbers under regular addition/multiplication. But here we are working over a different number system: $\mathbb{F}_2 = \{0,1\}$ under addition/multiplication modulo 2.

(b) Must there be a solution for every configuration $\boldsymbol{b}$?

(c) If not, what is the probability a random configuration $\boldsymbol{b}$ is solvable?

(d) When there is a solution for $\boldsymbol{b}$, is it unique? If not, can we find the smallest solution (i.e. giving the least number of button presses)?

24.2.1 Imagine you are in a field...

The algorithm learned in linear algebra for solving linear systems of the form $A\boldsymbol{x} = \boldsymbol{b}$ is known as Gauss-Jordan elimination (or simply as Gaussian elimination). The steps of the algorithm are as follows:

(a) Form the augmented matrix $[A|\boldsymbol{b}]$.

(b) Reduce the augmented matrix to *reduced row echelon form* by using elementary row operations:

- (swap) Swap any two rows.
- (scalar multiply) Multiply any row by a non-zero number.
- (replacement) Replace any row with a multiple of another row added to the row itself.

(c) Read off the solution (or conclude there isn't a solution) directly from the reduced row echelon form.

In linear algebra we only considered the real numbers $\mathbb{R}$ under addition/multiplication. If you were lucky you saw that the same thing could be done with complex numbers $\mathbb{C}$. We'd like to know, does all the theory developed in linear algebra carry over to more abstract sets of "numbers" under some sort of "addition" and "multiplication"? In particular what about the situation we are in with the lights out puzzle. Here our number system is

$$\mathbb{F}_2 = \{0, 1\}$$

and the addition and multiplication tables are defined as follows.

+	0	1
0	0	1
1	1	0

*	0	1
0	0	0
1	0	1

Does Gauss-Jordan elimination still work?

Let's consider a set F of objects which is closed under two operations $+$ and $*$. What properties would $(F, +, *)$ need to satisfy in order for Gauss-Jordan elimination to still possibly work?

First note the key to having this algorithm work is that the elementary row operations must be reversible. Clearly a row swap is reversible, just swap the rows back. Multiplying a row by a nonzero element a is reversible only if the element has a multiplicative inverse in F (a $b \in F$ such that $ab = 1$). Another key part to the algorithm is that we could use additive inverses to make entries of the matrix 0. This means we want our set of numbers F to satisfy the following 9 properties, known as the *field axioms*.

Definition 24.1: A set F with two operations $+$ and $*$ satisfying the following properties for every $a, b, c \in F$ is called a **field**.

(a) Addition is commutative, $a + b = b + a$.

(b) Addition is associative, $a + (b + c) = (a + b) + c$.

(c) There is a unique element 0 (zero) in F such that $a + 0 = a$.

(d) For each $a \in F$ there is a unique element $-a \in F$ such that $a + (-a) = 0$.

(e) Multiplication is commutative, $ab = ba$.

(f) Multiplication is associative, $a(bc) = (ab)c$.

(g) There is a unique element 1 (one) in F such that $a1 = a$

(h) For each non-zero $a \in F$ there is a unique element $a^{-1} \in F$ such that $aa^{-1} = 1$.

(i) Multiplication distributes over addition, $a(b + c) = ab + ac$ and $(b + c)a = ba + ca$.

So Gauss-Jordan elimination still works if our set of numbers F is a field. In our terminology of Chapter 10 properties (a)-(d) mean that F is an abelian group under $+$, and properties (e)-(h) mean that $F^* = F - \{0\}$ is an abelian group under $*$ (see Definition 10.1 for the definition of a group).

It turns out these were the only properties of $(\mathbb{R}, +, *)$ we used in linear algebra. Therefore, everything done in linear algebra holds true for matrices whose entries come from any field F.

Since $\mathbb{F}_2$ is a field with two elements then Gauss-Jordan elimination will work to solve the linear system. Moreover, any result we want to use from linear algebra will carry over to this new setting where our "numbers" come from $\mathbb{F}_2$.

24.2.2 Solving linear systems with Sage.

In a first course in linear algebra you were typically asked to solve linear systems by-hand. This was to allow you to understand the details of the Gauss-Jordan elimination algorithm. In practice, people don't generally solve systems of equations by hand, these are generally done by computer. We'll now see how to use Sage to solve linear systems.

To solve a linear system $Ax = b$ in Sage, we must first define the matrix A, for example

`matrix(ZZ,[[1,2],[3,4],[5,6]])` defines the matrix $\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$, over the integers $\mathbb{Z}$. Here we defined each row. We can give Sage a list and tell it how many rows, then have it split the list into a matrix as follows:

`matrix(QQ,2,[1,2,3,4,5,6])` defines the matrix $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$, over the rationals $\mathbb{Q}$,

Here is an example using Sage to solve the system

$$\begin{pmatrix} 1 & 0 & 2 \\ 3 & 2 & 5 \end{pmatrix} x = \begin{pmatrix} 3 \\ 0 \end{pmatrix}.$$

As we can see from the output of Sage the solution is $\begin{pmatrix} 3 \\ -9/2 \\ 0 \end{pmatrix}$.

Sage

```
sage: M=matrix(QQ,2,[[1,0,2],[3,2,5]])
sage: b=vector(QQ,[3,0])
sage: M.solve_right(b)          #command for solving Mx = b  (i.e. x is right of M)
(3, -9/2, 0)
```

The command for solving a linear system $A\boldsymbol{x} = \boldsymbol{b}$ is `A.solve_right(b)`. [1]

Coming back to the lights out puzzle, we first need to construct the lights out matrix A defined in (24.3). We could do it one entry at a time, which would involve entering $25 \cdot 25 = 625$ numbers. This wouldn't be fun, and if we want to consider larger game boards than 5×5 we would have a lot more typing to do. Instead, we use two loops to define A, and we do this for a general $n \times n$ board. Keep in mind, we have to tell Sage we are working over the field of integers modulo 2, $\mathbb{F}_2$. Sage knows this field by the name $GF(2)$, which stands for *Galois Field of size* 2.

Sage

```
sage: # Definition of the matrix for Lights Out
sage: # input = integer n (where lights out board is nxn)
sage: # output = lights out matrix A which is nxn
sage: def lights_out(n):
sage:    M = MatrixSpace(GF(2),n*n,n*n)    #tells SAGE to work with matrices in M_n(F_2)
sage:    A = M.matrix()      #initializes A to a matrix in M, we then define entries below
sage:    for i in range(n):
sage:        for j in range(n):
sage:            m = n*i+j
sage:            A[(m,m)] = 1
sage:            if i > 0 : A[(m,m-n)] = 1
sage:            if i < n-1 : A[(m,m+n)] = 1
sage:            if j > 0 : A[(m,m-1)] = 1
sage:            if j < n-1 : A[(m,m+1)] = 1
sage:    return A
```

For example the lights out matrix for the 3×3 game board is

Sage

```
sage: lights_out(3)

[1 1 0 1 0 0 0 0 0]
[1 1 1 0 1 0 0 0 0]
[0 1 1 0 0 1 0 0 0]
[1 0 0 1 1 0 1 0 0]
[0 1 0 1 1 1 0 1 0]
[0 0 1 0 1 1 0 0 1]
[0 0 0 1 0 0 1 1 0]
[0 0 0 0 1 0 1 1 1]
[0 0 0 0 0 1 0 1 1]
```

Asking for the lights out matrix for the 5×5 game returns confirmation it is stored in memory, but Sage saves us from having to look at it.

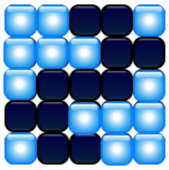
Sage

```
sage: lights_out(5)
25 x 25 dense matrix over Finite Field of size 2
```

[1] Using the word "left" would be the command to solve $\boldsymbol{x}A = \boldsymbol{b}$, but in this case $\boldsymbol{x}$ and $\boldsymbol{b}$ would be row vectors, not column vectors.

Now that A is loaded into Sage let's solve some configurations.

Example 24.1: Solve the following configuration:



The configuration matrix is $B = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \end{pmatrix}$ which we can express as a vector

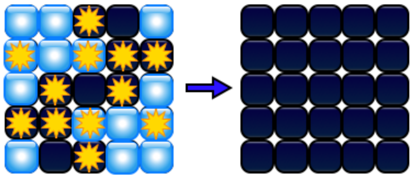
$$\boldsymbol{b} = (1,1,0,0,1,\ 1,1,1,0,0,\ 1,0,0,0,1,\ 0,0,1,1,1,\ 1,0,0,1,1).$$

(Spaces are inserted after each group of 5 entries in $\boldsymbol{b}$ so it is easier to read.) Now we have Sage solve $A\boldsymbol{x} = \boldsymbol{b}$.

Sage

```
sage: #current game configuration (i.e. buttons that are lit)
sage: b=vector(GF(2),[1,1,0,0,1, 1,1,1,0,0, 1,0,0,0,1, 0,0,1,1,1, 1,0,0,1,1]);

sage: #solving the game
sage: x=lights_out(5).solve_right(b);

sage: #now put the solution x in a nice matrix form so we can see what buttons to press
sage: button_press_matrix = matrix(GF(2),5,5,x.list())     # convert vector to a matrix
sage: button_press_matrix                                  # show matrix in output

[0 0 1 0 0]
[1 0 1 1 1]
[0 1 0 1 0]
[1 1 1 0 1]
[0 0 1 0 0]
```

Therefore, to solve the puzzle we just need to press the 12 buttons shown in the diagram below.



Rather than have to type out the previous lines of code every time we want to solve a configuration we could build a *solve* function as follows:

Lights-Out Solve function: (basic version)

Sage

```
sage: # Definition of the solution function for Lights Out
sage: # input = integer n (where lights out board is nxn), and b the configuration vector
```

```
sage: # output = a matrix X indicating which buttons to press for solution
sage: def lights_out_solver(n,b):
sage:     x=lights_out(n).solve_right(b);
sage:     button_press_matrix = matrix(GF(2),n,n,x.list())
sage:     return button_press_matrix
```

For our previous example we could just type:

Sage

```
sage: b=vector(GF(2),[1,1,0,0,1, 1,1,1,0,0, 1,0,0,0,1, 0,0,1,1,1, 1,0,0,1,1]);
sage: lights_out_solver(5,b)
[0 0 1 0 0]
[1 0 1 1 1]
[0 1 0 1 0]
[1 1 1 0 1]
[0 0 1 0 0]
```

24.2.3 Solvable Configurations

A lit button configuration $\boldsymbol{b}$ is solvable if the corresponding linear system $A\boldsymbol{x} = \boldsymbol{b}$ has a solution. From linear algebra we know

$$A\boldsymbol{x} = \boldsymbol{b} \text{ is solvable for \textit{every} } \boldsymbol{b} \iff A \text{ is invertible} \iff \det(A) \neq 0.$$

The lights out matrix (for $5{\times}5$ game) has determinant 0. Therefore, there do exist unsolvable configurations $\boldsymbol{b}$.

Sage

```
sage: lights_out(5).determinant()
0
```

For example, the configuration in Figure 24.5 is unsolvable.

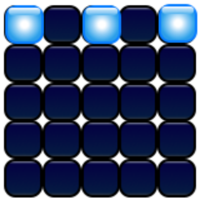


Figure 24.5: An unsolvable configuration of lights.

Sage

```
sage: b=vector(GF(2),[1,0,1,0,1, 0,0,0,0,0, 0,0,0,0,0, 0,0,0,0,0, 0,0,0,0,0]);
sage: lights_out_solver(5,b)

Traceback (click to the left of this block for traceback)
...
ValueError: matrix equation has no solutions
```

Recall that $A\boldsymbol{x} = \boldsymbol{b}$ has a solution only when $\boldsymbol{b}$ is in the column space of A, denoted col(A). This is just a fancy way of saying

$$\sum_{1 \leq i,j \leq 5} x_{i,j}\boldsymbol{t}_{i,j} = \boldsymbol{b}$$

for some $x_{i,j}$, where $\boldsymbol{t}_{i,j}$ are the toggle vectors, as we already know. However, phrased in this way we see that the set of solvable configurations is $\text{col}(A) = \text{span}(\boldsymbol{t}_{1,1}, \boldsymbol{t}_{1,2}, \ldots, \boldsymbol{t}_{5,5})$, and the dimension of $\text{col}(A)$ is called the rank of A, denoted $\text{rank}(A)$.

Sage

```
sage: lights_out(5).rank()
23
```

Therefore only 23 buttons are required to solve any configuration, and if each one can either be pressed or not, then there are 2^{23} solvable configurations, out of a possible 2^{25} configurations. This proves the following theorem.

Theorem 24.1: *For the* 5×5 *lights out puzzle, the probability that a random configuration is solvable is* $1/4$.

Quiet Patterns:

There exist sequences of button presses that will leave the lights unchanged. These are known as **quiet patterns**. Such a sequence $\boldsymbol{x}$ is a solution to the homogeneous equation $A\boldsymbol{x} = \boldsymbol{0}$. That is, $\boldsymbol{x}$ is in the null space of A, denoted by $\text{nul}(A)$. The dimension of this space is $\text{nullity}(A) = 25 - \text{rank}(A) = 25 - 23 = 2$. If we let $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ be a basis for $\text{nul}(A)$ then

$$\begin{aligned}\text{nul}(A) &= \text{span}(\boldsymbol{d}_1, \boldsymbol{d}_2) = \{r_1\boldsymbol{d}_1 + r_2\boldsymbol{d}_2 \mid r_1, r_2 \in \mathbb{F}_2\} \\ &= \{\boldsymbol{0}, \boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_1 + \boldsymbol{d}_2\}.\end{aligned}$$

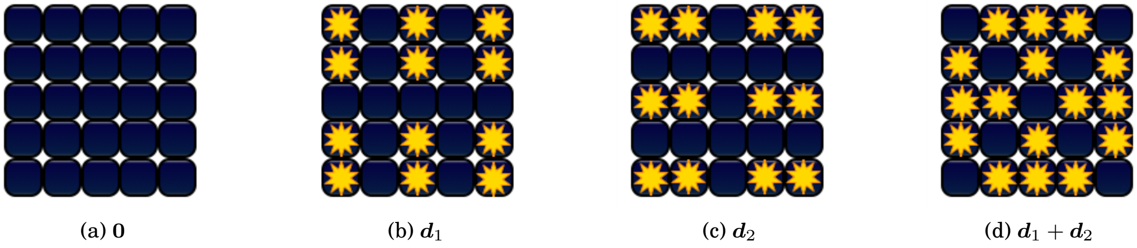
Therefore, there are only 4 such button sequences (vectors).

We can use Sage to find these vectors. The command for computing the null space is `.right_kernel()`.

Sage

```
sage: lights_out(5).right_kernel()
Vector space of degree 25 and dimension 2 over Finite Field of size 2
Basis matrix:
[1 0 1 0 1 1 0 1 0 1 0 0 0 0 0 1 0 1 0 1 1 0 1 0 1]
[0 1 1 1 0 1 0 1 0 1 1 1 0 1 1 1 0 1 0 1 0 1 1 1 0]
```

Sage returns a basis for the nullspace. The span of these vectors (using coefficients from $\mathbb{F}_2 = \{0, 1\}$) gives us the complete null space. These correspond to the button presses shown in Figure 24.6.



(a) $\boldsymbol{0}$ (b) $\boldsymbol{d}_1$ (c) $\boldsymbol{d}_2$ (d) $\boldsymbol{d}_1 + \boldsymbol{d}_2$

Figure 24.6: The 4 Quiet Patterns: These are the button press sequences in the nullspace of A. Starting with all the lights out and pressing the buttons indicated in any of these four patterns will result in all the lights being out again.

24.2.4 Optimal solution to Lights Out

Let $\boldsymbol{b}$ be a (solvable) configuration of the lights. If $\boldsymbol{x}$ is a strategy vector (i.e. a solution to $A\boldsymbol{x} = \boldsymbol{b}$) then the set of *all* solution strategies is: [2]

$$\boldsymbol{b} + \text{null}(A) = \{\boldsymbol{b}, \quad \boldsymbol{b} + \boldsymbol{d}_1, \quad \boldsymbol{b} + \boldsymbol{d}_2, \quad \boldsymbol{b} + \boldsymbol{d}_1 + \boldsymbol{d}_2\}.$$

The optimal solution will be the one with the fewest number of 1's as entries.

Let's go back to Example 24.1 and see if we can find an optimal solution. The one we found requires 12 button presses, perhaps we can do better.

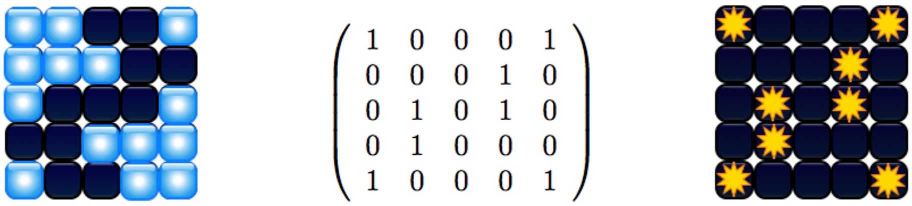
It will be convenient to have Sage count the number of occurrences of 1 in a strategy vector. We will define a function called `number_of_presses` to do this.

```
Sage
sage: def number_of_presses(x):
sage:     counter=0;    # initialize counter, which is our variable to count 1's
sage:     for i in range(0,25):      # recall Python indexes lists from 0, not 1
sage:        if x[i]==1: counter=counter+1    # check if ith entry is 1, increment counter
sage:    return counter
```

Now let's find all 4 solutions to Example 24.1.

```
Sage
sage: b = vector(GF(2),[1,1,0,0,1, 1,1,1,0,0, 1,0,0,0,1, 0,0,1,1,1, 1,0,0,1,1]);
sage: x = lights_out(5).solve_right(b)    # one solution
sage: nulsp = lights_out(5).right_kernel()
sage: for d in nulsp:
sage:    print b+d, number_of_presses(b+d)
(0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0)  12
(1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1)  8
(0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0)  8
(1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1)  20
```

There are two optimal solutions, each requiring 8 button presses. Therefore, an optimal solution to the configuration in Figure 24.7a is the strategy matrix in Figure 24.7b.



$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

(a) Configuration of lights

(b) optimal strategy matrix

(c) optimal solution

Figure 24.7: An optimal solution requiring 8 button presses.

[2] For a set A and element x the notation $x + A$ represents the set obtained by adding x to every element of the set A. That is, $x + A = \{x + a \mid a \in A\}$.

24.3 Summary of 5×5 lights out puzzle

Solving a configuration $\boldsymbol{b}$ of the lights out puzzle is equivalent to solving the linear system $A\boldsymbol{x} = \boldsymbol{b}$ for strategy vector $\boldsymbol{x}$ where A is a 25×25 lights out matrix. All arithmetic is done in the finite field of size 2: $\mathbb{F}_2 = \{0, 1\}$.

- The dimension of space of solvable configurations is rank$(A) = 23$, the number of solvable configurations is $2^{\text{rank}(A)} = 2^{23}$.
- The probability that a random configuration is solvable is $2^{23}/2^{25} = 1/4$,
- The dimension of space quiet patterns (nul(A)) is nullity$(A) = 5^2 -$ rank$(A) = 2$, the number of quiet patterns is $|\text{nul}(A)| = |\mathbb{F}_2|^{\text{nullity}(A)} = 2^2 = 4$.
- For a given strategy vector $\boldsymbol{x}$ the 4 equivalent vectors are the elements of $\boldsymbol{x} + \text{nul}(A)$.

Putting all the previous ideas into one code block, we can write a lights out solver which returns the optimal solution.

Lights-Out Solve function: (optimal version)

Sage

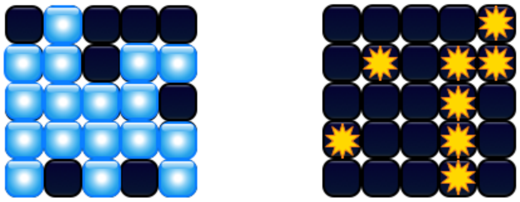
```
sage: # Function:  number_of_presses
sage: # input = a vector x of dimension 25 with 0,1 entries
sage: # output = the number of times 1 appears as an entry
sage: def number_of_presses(x):
sage:     counter=0;
sage:     for i in range(0,25):
sage:         if x[i]==1: counter=counter+1
sage:     return counter

sage: # Function:  optimal_solution
sage: # input = a strategy vector x
sage: # output = an equivalent strategy vector which uses least number of button presses
sage: def optimal_solution(x):
sage:     op_button_presses=x     # initialize variable to store optimal solution
sage:     n=number_of_presses(x)  # initial variable to store optimal presses
sage:     nul=lights_out(5).right_kernel()
sage:     for d in nul:
sage:         if number_of_presses(x+d)<n:
sage:             op_button_presses=x+d        # update variable
sage:            n=number_of_presses(x+d)      # update variable
sage:    return op_button_presses

sage: # Function:  lights_out_solver
sage: # input = b the configuration vector of lights on 5-by-5 game
sage: # output = an optimal strategy matrix which solves the puzzle
sage: def lights_out_solver(b):
sage:     x=lights_out(5).solve_right(b);  # one solution
sage:     x=optimal_solution(x)     # exchanges x for an optimal solution
sage:     button_press_matrix = matrix(GF(2),5,5,x.list()) #turn output vector into matrix
sage:     return button_press_matrix
```

As an example, to solve the configuration in Figure 24.8a we proceed as follows.

Sage

```
sage: b=vector(GF(2),[0,1,0,0,0, 1,1,0,1,1, 1,1,1,1,0, 1,1,1,1,1, 1,0,1,0,1]);
sage: lights_out_solver(b)
```

```
[0 0 0 0 1]
[0 1 0 1 1]
[0 0 0 1 0]
[1 0 0 1 0]
[0 0 0 1 0]
```



(a) Configuration of lights

(b) an optimal solution

Figure 24.8: An optimal solution requiring 8 button presses.

24.4 Eigenvalues and Eigenvectors

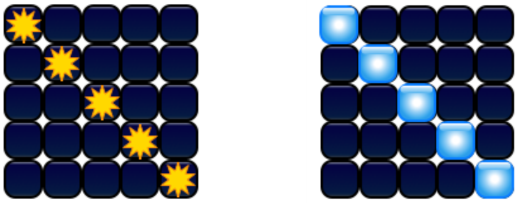
Here we discuss how the lights out puzzle gives an example of how to visualize eigenvalues and eigenvectors, which is a topic covered in a first course in linear algebra.

Recall that $\boldsymbol{x}$ is an *eigenvector* of A if $x \neq 0$ and

$$A\boldsymbol{x} = \lambda \boldsymbol{x}$$

for some scalar λ. The scalar λ is called the corresponding *eigenvalue*. In our case we only have two scalars: 0 and 1, so either (i) $A\boldsymbol{x} = \mathbf{0}$ or (ii) $A\boldsymbol{x} = \boldsymbol{x}$. We already considered $A\boldsymbol{x} = \mathbf{0}$ (which corresponds to the nullspace, and what we called the quiet patterns). So let's consider $A\boldsymbol{x} = \boldsymbol{x}$, the space of eigenvectors corresponding to eigenvalue 1.

A solution to $A\boldsymbol{x} = \boldsymbol{x}$ would be a pattern such that if, starting with all the lights off, then pressing the buttons corresponding to the solutions, exactly the pressed lights will end up being on. For an example see Figure 24.9.



(a) Button presses corresponding to an eigenvalue

(b) Light configuration after pressing buttons in 24.9a

Figure 24.9: An example of an eigenvector for the lights out matrix.

How many of these patterns are there? That is, how many eigenvectors are there which correspond to the eigenvalue 1? The solutions to $A\boldsymbol{x} = \boldsymbol{x}$ are precisely the solutions to $(A - I)\boldsymbol{x} = \mathbf{0}$, where I is the 25×25

identity matrix. So the answers lies with the nullspace of $A - I$. From the calculation below we see the dimension of the nullspace of $A - I$ is 5.

Sage
```
sage: I25=MatrixSpace(GF(2),25,25).identity_matrix();      # 25x25 identity matrix
sage: A=lights_out(5);
sage: (A-I25).nullity()

5
```

Therefore, there are 5 linearly independent eigenvectors (with $\lambda = 1$) which form a basis for nul$(A-I)$. Any linear combination of these five vectors, with coefficients 0 or 1, is an eigenvector. This gives 2^5 possibilities (including $\mathbf{0}$ which is not a eigenvector but does solve $A\boldsymbol{x} = \boldsymbol{x}$).

In Exercise 4 you are asked to find 5 vectors which form a basis for the eigenspace corresponding to $\lambda = 1$.

24.5 Other sized game boards

Lights Out has be modified and generalized in many ways: bigger games boards, different toggle conditions, more states (colours) for the lights to cycle through.

Here we mention briefly some results about larger games boards. We assume the toggling condition is the same as for the 5×5 game board. Let A_n be the lights out matrix for the $n \times n$ game board. The key to understanding solvability lies in knowing the whether the $n^2 - \text{rank}(A_n)$ is 0 or not. If it is 0 then A_n has full rank, and so it's columns are linearly independent, therefore A_n is invertible. This means *every* configuration is solvable and has a unique solution in which no button is pressed more than once. If it is non-zero then $\text{rank}(A_n) < n^2$ so A_n is not invertible, therefore there exist configuration which are not solvable. Moreover, the number of different solutions for a given configuration (if the configuration is solvable) is $2^{\text{nullity}(A_n)} = 2^{n^2 - \text{rank}(A_n)}$.

Table 24.1 lists the values of $n^2 - \text{rank}(A_n)$ for $3 \leq n \leq 10$.

n	rank(A_n)	nullity$(A_n) = n^2 - \text{rank}(A_n)$
3	9	0
4	12	4
5	23	2
6	36	0
7	49	0
8	64	0
9	73	8
10	100	0

Table 24.1: nullity(A_n) and rank(A_n) for various boards sizes of lights out.

24.6 Light-Chasing Strategy

There is a strategy for solving the 5×5 lights out puzzle which, though not optimal, will allow you to solve the puzzle without having to solve a linear system. The technique is known as *light-chasing*. Begin with the top row and press the button beneath any lit button in the top row. This will turn out all lights in the top row. Apply this strategy row by row until you reach the bottom row.

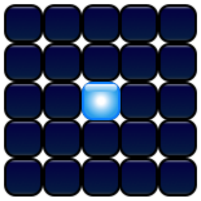
The lights in the bottom row will be one of the 7 configurations shown in Table 24.2, press the corresponding buttons in the top row as indicated in the table. Then apply the light-chasing strategy again, beginning from the top row. This will solve the puzzle. Can you use the techniques we discussed in this lecture to come up with the button press sequences shown in the right-hand column of the table?

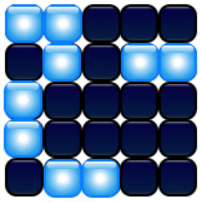
Lights on bottom row$n$	Press these on top row
🟦■■■🟦	✸✸■■■
■🟦■🟦■	✸■■✸■
🟦🟦🟦■■	■✸■■■
■■🟦🟦🟦	■■■✸■
🟦■🟦🟦■	■■■■✸
■🟦🟦■🟦	✸■■■■
🟦🟦■🟦🟦	■■✸■■

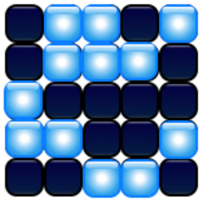
Table 24.2: Light-chasing strategy

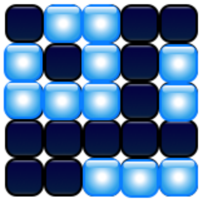
24.7 Exercises

1. Solve each of the following configurations.
   (You can use the Lights-out puzzle on Jaap's puzzle page to edit the lights, then try out your solution.)

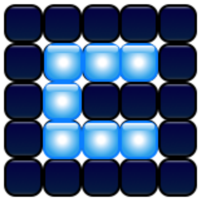

(a)

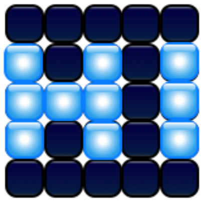

(b)

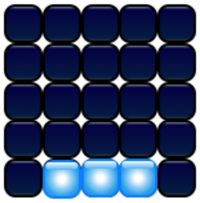

(c)


(d)

2. Show each of the following configurations are not solvable.


(a)


(b)


(c)

3. In Table 24.2 there are 7 cases shown for the configuration of lights in the last row of the puzzle. The case in which all the lights in the last row are out isn't included in this table since the puzzle would already be solved in this case. Show that the other $2^5 - 8 = 24$ configurations are not solvable. This explains why they are not considered in the table.

4. In Section 24.4 we discussed how to visualize eigenvectors on the lights out puzzle. In this exercise find five linearly independent vectors which are eigenvectors corresponding to the eigenvalue 1. Once you found these vectors verify them visually by starting with all the lights off, then pressing the buttons corresponding to the vector observe that exactly the pressed lights will end up being on.

5. Consider a variant of the lights out game on an $n \times n$ board in which a button press changes the state of the button itself and its four *diagonal* neighbours. All other rules remain the same.

   (a) What is the lights out matrix for this game when $n = 4$?

   (b) For $n = 3, 4, \ldots, 8$ calculate the dimension of the space of solvable configurations and the dimension of the space of quiet patterns.

   (c) Compare your answers to the situation for the classic toggle rule which are shown in Table 24.1.

6. Consider the variation of the game in which each light has 3 states (red, green, off). You can assign to each state a number in the field $\mathbb{F}_3 = \{0, 1, 2\}$. Assume the classic toggle rules apply (only the button pressed and its horizontal and vertical neighbours are toggled). Explain what the elements of each eigenspace represent in terms of the game.

Appendix A

Sage

Ted Kosen, author of *Sage For Newbies*, describes Sage as follows

> Sage is an open source mathematics computing environment for performing symbolic, algebraic, and numerical computations. Mathematics computing environments are complex, and require a significant amount of time and effort to become proficient at using one. It will take a beginner a while to become an expert using Sage, but fortunately one does not need to be a Sage expert in order to begin using it to solve problems.

This is precisely the viewpoint we will take in this book. We will not attempt to become Sage experts, we will however us it to solve problems. In particular, problems regarding permutation puzzles.

A mathematics computing environment is a collection of computer algorithms, and data structures, that are built on top of a programming language. This means one has access to a full programming language, in Sage's case it is Python, and further access to a mathematical objects library complete with algorithms for performing calculations.

There is an excellent online Sage tutorial at http://www-rohan.sdsu.edu/∼mosulliv/sagetutorial/. It covers everything we'll discuss in this appendix plus a whole lot more.

Rather than say anything more about what Sage is, let's just see for ourselves what it can do.

A.1 Sage Basics

We'll be using Sage through an online server via a web browser (called a "Sage notebook"). This means you can use it through your desktop computer or mobile device such as a laptop or iPad. It is not necessary to download and install the (free) Sage software package, however you are free to do so if you want your own local copy. See [16] for more information.

Once you log-in to the Sage server (first time requires you to create a free account) a worksheet manager page will be displayed. See Figure A.1. Here you can create new worksheets or continue to work with previously created worksheets.

To create a new worksheet click the "New Worksheet" link. You will be prompted to give the worksheet a title (by default the worksheet is given the name "Untitled"). A blank worksheet will be displayed (see Figure A.2). You can now begin working in Sage by typing commands in the cell (code-clock) where I've written "type code here".

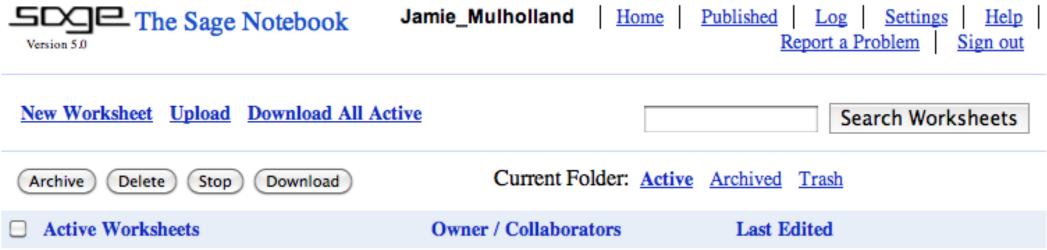


Figure A.1: Sage worksheet management page.

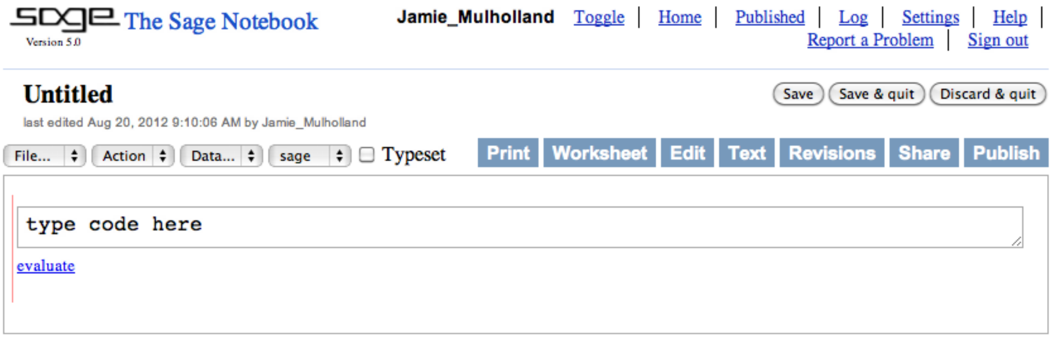


Figure A.2: Sage blank worksheet.

Let's now have a look at the kinds of computations we can do in Sage. In this book we will display Sage cells as shown below in order to make the text easier to read.

Sage
```
sage: input (or source code) will be typed here
output will be displayed here
```

A.1.1 Arithmetic Operations

We can use Sage like a calculator to add/subtract/multiply/divide numbers.

Sage
```
sage: 2+3
5
```

When typing input, [enter] will jump you down to the next line, whereas [shift-enter] gets Sage to evaluate the code-block. You can also press the [evaluate] link below the active cell instead of [shift- enter].

A bit of terminology: What is typed in a cell is called the *source code*. When the cell is executed, what Sage prints to the screen (blue writing in the Sage notebook) is called the *output*.

So "2+5" in the cell above is the source code, and "5" is the output.

Common arithmetic operations are:

operation	syntax name
addition	+
subtraction	−
multiplication	*
division	/
remainder	%
integer quotient	//
exponentiation	^ or **

Sage

```
sage: 3+2/5*2^3
31/5
sage: 2^3
8
sage: 2**3
8
sage: (5^3)^2
15625
sage: 11%4
3
sage: 7//3
2
```

As we can see Sage follows the usual order of operations:

(a) first evaluate exponents from right to left,

(b) then multiplication, division, remainder from left to right,

(c) finally, addition and subtraction from left to right.

The order in which expressions are evaluated can be changed using parenthesis: ( ).

A.1.2 Inserting a New Cell

A new cell will automatically appear below the last cell of your sheet when the contents of the last cell have been evaluated. Sometimes, however, we would like to add a new cell in the middle of our worksheet. To insert a new execution shell in the worksheet, you can:

Hover the mouse slightly above or below the current cell, until a purple horizontal line appears, then click the line. This will add a cell where you clicked.

If you just want to add a cell, or a bunch of cells, at the end of the worksheet, just select the last cell and hit [shift-enter] to add a new cell.

A.1.3 Working in a Cell: Enter, Shift-Enter, The Semicolon, and Comments

Multiple statements can be placed in a single cell. After typing ”1+2”, use [enter] to bring the cursor down to the next line. Notice when the cell is evaluated (either [shift-enter] or click [evaluate] below cell) only the output of the last line of code is displayed.

Sage

```
sage: 1+2
sage: 3-2
1
```

To display both outputs we can place the statements on the same line, separating them by semicolons ”;”. In Sage, semicolons can be placed after statements as optional terminators, but most of the time one will only see them used to place multiple statements on the same line.

Sage

```
sage: 1+2;  3-2;
3
1
```

There is also a ”print” command for sending outputs to the screen. We’ll see this in Section A.2.2.

It will come in handy to be able to add comments directly in the source code. Comments are basically notes to yourself about what you are doing, and are not intended for the computer to execute. Anything followed by the ”#” symbol is treated as a comment.

Sage

```
sage: # this is a comment
sage: 2-3 # and this is another comment after some code
-1
```

When a comment is too long to fit in one line, we can enclose it in triple quotes: ””” text here ”””.

Sage

```
sage: """here is a really long comment. What the next line does is evaluates 3
expressions all on the same line. Nothing special, but my comment is now pretty long."""
sage: 5^4; 4+2*3;
625
10
```

A.2 Variables and Statements

Part of the power of a computing environment lies in the ability to store, manipulate, and recall information. This is done using ”variables” and ”statements”.

A.2.1 Variables

A **variable** is a name that is associated with the data stored in a memory address. One way to create variables in Sage is through **assignment** which consists of placing the variable you would like to create to the left of an equal sign ”=”, and the expression on the right side.

Here we create a variable ”a”, and assign to it the number 5.

Sage

```
sage: a=5      # create a new variable a and assign 5 to it.
sage: b=7      # create a new variable b and assign 7 to it.
sage: a=3      # reassign to the variable a the number 3.
sage: c=a+b    # assign to the variable c the sum of a and b
sage: c        # output the value of c
10
```

A.2.2 Statements

Statements are the part of a programming language that are used to encode algorithmic logic.

Simple statements:

- **assignment**:
  a=a+1
- **call**:
  var(x), print(a+1), factor(24)
- **assumption**:
  assume($x > 0$)

Compound statements:

- **if-statement**:
  ```
  if $A > 3$:
      print(A-3)
  else:
      print(”not big enough”)
  ```
- **while statement**:
  ```
  while $x <= 10$:
      print(x)
      x=x+1
  ```
- **for-statement**:
  ```
  for x in [1,2,3,4,5]:
      print x
  ```

We will look at *if*, *while*, and *for* statements more thoroughly in Section A.6. But the odd one may creep into some of our examples below.

print()

Sage has a statement called ”print” that allows the results of expressions to be displayed regardless of where they are located in the cell.

Sage

```
sage: a=3
sage: b=2
sage: print a,b
3  2
```

We could also have used semicolons to get a and b on the same line. Notice though, this displays a and b on separate lines in the output, whereas ”print” puts them on the same line.

Sage

```
sage: a=3; b=2;
sage: a; b;
3
2
```

Sage

```
sage:  if is_prime(5):           # is_prime() is a built-in function
sage:      print("5 is prime")
sage:  else:
sage:      print("5 is not prime")
5 is prime
```

A.3 Lists

Lists are one of the most fundamental objects in any programming language.

A.3.1 Defining a List

A list is defined by putting the items of the list, separated by commas, inside square brackets ”[ ]”.

Sage

```
sage: L=[1,2,"milk","cheese","new shoes"]
sage: print L
[1, 2, ‘milk’, ‘cheese’, ‘new shoes’ ]
```

We can select items from the list as follows. (Note: the first item in a Sage/Python list is indexed by 0, not 1 as you may have expected.)

Sage

```
sage: L[0]    # select the first item (indexed by 0) from the list L
1

sage: L[4]    # select the last item
‘new shoes’

sage: L[5]    # what if there is no item with the index you’ve input
Traceback (click to the left of this block for traceback)
...
IndexError: list index out of range
```

Order matters in a list. If items are in different orders, then the list are not equal.

Sage

```
sage: [1,2,3]==[2,1,3]
False
```

We can also create a list by stating conditions we want the elements to satisfy. This usually requires starting with a bigger list, and either constructing a sublist, or constructing a new list.

Sage

```
sage: [n for n in [1,2,3,4] if is_even(n)]   # selects the even integers
[2,4]

sage: [2*n+1 for n in [1,2,3,4]]   # creates a new list of odd integers from the old list
[3,5,7,9]
```

A.3.2 List Operations: remove, append, etc.

Sage

```
sage:  L=[1,2,3]
sage:  L.remove(3)   # removes item 3 from L
sage:  L
[1,2]
sage:  L.append(4)   # adds item 4 to the end of list L
sage:  L
[1,2,4]
```

```
sage:  len(L)
3
```

Here are some more list operations. We won't really use any of them, but just in case you are curious. In each of the following ”L” is used as the name of our list and the operation being described is typed in bold.

operation	**description**
x **in** L	True if item x is in list L, else False.
x **not in** L	False if item x is in list L, else True.
L **+** S	The concatenation of lists L and S.
n*L, or L***n**	n copies of list L concatenated.
L**[i]**	ith entry of list L (first entry has index 0).
L**[i:j]**	slice of L from i up to, but not including j.
L**[i:j:k]**	slice of L from i to j with step k.
len(L)	length of list L.
min(L)	smallest item in L.
max(L)	largest item in L.
L.**append**(x)	Add an item x to the end of the list L
L.**extend**(S)	Extend the list L by appending all the items in the given list S to the end of L
L.**insert**(i,x)	Insert item x in position i of list L. For example, L.insert(0, x) inserts x at the front of the list, and L.insert(len(L), x) is equivalent to L.append(x).
L.**remove**(x)	Remove the first item from the list whose value is x. It is an error if there is no such item.
L.**pop**(i)	Remove the item at the given position in the list, and return it. If no index is specified, L.pop() removes and returns the last item in the list.
L.**index**(x)	Return the index of the first occurrence of item x in the list. It is an error if there is no such item.
L.**count**(x)	Return the total number of times item x appears in the list.

A.3.3 Built-In Lists

Sage (and Python) already have some built-in lists that we can use. The range function, `range(a,b)`, creates the list of integers beginning with a and ending at b-1. Here we assume a is less than b.

Sage

```
sage:  range(1,11)   # creates a list of integers from 1 to 10
[1,2,3,4,5,6,7,8,9,10]

sage:  [1..10]    #another way to construct the list
[1,2,3,4,5,6,7,8,9,10]
```

We can select one or multiple items from a list as follows.

Sage

```
sage:  L=range(1,26)
sage:  print L[1]      # selects item of index 1
sage:  print L[0:3]   # sublist of L consisting of items indexed 0 through 3
```

```
2
[1,2,3]
```

The items in a list can be overwritten by new items, as the following shows.

Sage

```
sage:  L=[1,2,3,4]
sage:  L
[1,2,3,4]
sage:  L[1]=5
sage:  L
[1,5,3,4]
```

An example of using lists to factor all integers from 1 to 10.

Sage

```
sage:  [factor(n) for n in range(1,11)]
[1, 2, 3, 2^2, 5, 2 * 3, 7, 2^3, 3^2, 2 * 5]
```

A.4 Sets

Sage has a built-in Set type. It offers a fast lookup of whether an element is in the set or not, and it comes equipped with standard set-theoretic operations: union, intersection, etc.

Unlike lists, where order matters, the order of the elements in a set does not matter. All the matters is the elements themselves. So in this sense, we can think of sets as unordered lists.

A.4.1 Defining a Set

Sage

```
sage:  Set([1,2,3])        # here we turn the list [1,2,3] into a set
{1,2,3}

sage:  Set([1,2,3])==Set([2,1,3])        # order doesn't matter in a set
True

sage:  Set(range(1,101))        # using the range list to construct a set
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24, 25,
26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 52, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 71, 72,
73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 88, 89, 90, 91, 92, 93, 94, 95,
96, 97, 98, 99, 100}
```

Another way to construct the same set.

Sage

```
sage:  Set(1..100)
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24, 25,
26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 52, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 71, 72,
73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 88, 89, 90, 91, 92, 93, 94, 95,
96, 97, 98, 99, 100}
```

We can construct sets by specifying conditions on the elements, much like we did with lists.

Sage

```
sage:  Set(x for x in range(1,11) if is_even(x))
{8,2,4,10,6}

sage:  Set(x for x in range(1,51) if x%3==0)        # integers divisible by 3
{33, 3, 36, 6, 39, 9, 42, 12, 45, 15, 48, 18, 21, 24, 27, 30}
```

There is also a filter command for selecting elements satisfying some special condition. Here we select the prime numbers.

Sage

```
sage:  S=Set(1..20)
sage:  filter(is_prime,S)
[2, 3, 5, 7, 11, 13, 17, 19]
```

See Lecture 2 for some further examples of sets in Sage, including the set-theoretic operations: union, intersection, etc.

A.5 Commands/Functions

We have already seen a few built-in Sage/Python commands: `print()`, `is_prime()`, `factor()`, etc. We now look at two ways to define our own commands.

A.5.1 Defining Your Own Commands

The syntax for defining a command, which in this template we have called "function_name", is:

```
def function_name( <parameters> ):
         :
    <statement>
    <statement>
         :
    return <expression>
```

Here are some examples.

Sage

```
sage:  def f(x):
sage:      return x*x

sage:  f(2)   # here we test the command f
4
sage: f(7)
49
```

Sage

```
sage:  def is_divisible_by_three(x):
sage:       if x%3==0:      # check to see if the remainder is 0 when divided by 3
sage:           return True
sage:       else:
sage:           return False        # this ends the definition of the command
```

```
sage:  is_divisible_by_three(6)    # here we test the command
True
sage: is_divisible_by_three(7)
False
```

Commands can take more than one parameter.

Sage

```
sage:  # this function capitalizes the word x then concatenates it with itself n times
sage:  def repeat_word(x,n):
sage:      return x.capitalize()*n

sage:  repeat_word("limabean",3)
’LimabeanLimabeanLimabean’
```

A.5.2 Lambda Functions

Python supports the creation of anonymous functions (i.e. functions that are not bound to a name) using a construct called ”lambda”. This is a very powerful concept that’s well integrated into Python and is often used in conjunction with typical functional concepts like `filter()` and `map()`.

Using the lambda function we can create a function like f above.

Sage

```
sage:  f = lambda x: x*x
sage:  f(2)
4
sage:  f(7)
49
```

A lambda function can take more than one argument.

Sage

```
sage:  concat = lambda x,y: x+y
sage:  concat("super","man")
superman
```

Also note that you can put a lambda definition anywhere a function is expected, and you don’t have to assign it to a variable at all. Here we use a lambda function along with `filter()` to pick out all elements of a list which are divisible by 3.

Sage

```
sage:  print filter(lambda x: x%3==0,range(1,50))
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48]
```

A.6 *if*, *while*, and *for* statements

A.6.1 *if* statement

A *conditional statement* is what we use when we want our code to make decisions. For example, suppose we wanted to divide a number by 2 only if it is even. We can do this in Sage by using an `if` statement. The general syntax for Python’s `if-else` statement is

```
if <condition>:
    <statement>
         :
else:
    <statement>
         :
```

`<condition>` is an expression that can use *relational operators* <, >, <=, >=, == (is equal), and != (is not equal), as well as *logical operators*: `and`, `or`, `not`. Its value is either True or False. If the condition is true, the statements indented under `if` are executed, otherwise the statements under `else` are executed. The `else` clause is optional: you can have `if` alone. In that case, if the condition is true the program executes the statements indented under `if`, otherwise the program skips them.

Sage

```
sage:  n = 15
sage:  if n%3==0:
sage:      print(’n is divisible by 3’)
sage:  else
sage:      print(’n is not divisible by 3’)
n is divisible by 3
```

Often you'll need to string a chain of several `if-else` statements together. For example

Sage

```
sage:  def letterGrade(score):
sage:      if score >= 90:
sage:          return ’A’
sage:      else:
sage:          if score >= 80:
sage:              return ’B’
sage:          else:
sage:              if score >= 70:
sage:                  return ’C’
sage:              else:
sage:                  if score >= 60:
sage:                      return ’D’
sage:                  else:
sage:                      return ’F’
```

Python lets you simplify the indentation and compress the ”if-else” on one line by using the keyword `elif`. The above code can be shortened to

Sage

```
sage:  def letterGrade(score):
sage:      if score >= 90:
sage:          return ’A’
sage:      elif score >= 80:
sage:          return ’B’
sage:      elif score >= 70:
sage:          return ’C’
sage:      elif score >= 60:
sage:          return ’D’
sage:      else:
sage:          return ’F’
```

A.6.2 *while* loop

while loops are one of the most useful techniques in programming. Essentially, a while loop allows us to repeat the same block of statements multiple times (but with different values of variables) while a certain condition holds true. The general syntax for Python's while loop is

```
while <condition>:
    <statement>
         :
```

As long as the condition remains true the program repeats the statements in the while block.

The next example uses a while loop to add the integers from 1 to 10.

Sage

```
sage: i = 1
sage: sum1ton = 0
sage: while i <= 10:
sage:     sum1ton += i    # equivalent to sum1ton = sum1ton + i
sage:     i += 1          # increments i by 1
sage: sum1ton
55
```

A.6.3 *for* loop

for loops are traditionally used when you have a block of code which you want to repeat a fixed number of times. In Python, for loops iterate over a fixed list. As an alternative, the while loop could be used, however, while is used when a condition is to be met, or if you want a block of code to theoretically repeat forever, for example repeatedly asking for user input until the format the user provides is correct. The general syntax for Python's for loop is

```
for  x in <list>:
    <statement>
         :
```

Here is an example of using the for loop to step through the entries of a list and square each one.

Sage

```
sage:  L = [1,2,3,4,5]
sage:  for x in L:
sage:      print x^2
1
4
9
16
25
```

In the next example we use a for loop together with an if statement to print the list of integers from 1 to 20 which are divisible by 3.

Sage

```
sage:  for x in range(1,21):
sage:      if x%3 == 0:
sage:         print x
```

```
3
6
9
12
15
18
```

This has been a quick introduction to get you up and running with Sage. We will be developing our experience with Sage throughout this book, so have fun!.

A.7 Exercises

1. Assign 27 to the variable a, 1027 to the variable b, and the product to the variable c. Have Sage output the values of all three variables.
2. For a, b and c in Exercise 1, use the factor() command to factor c. Also, use the remainder command % to determine if 16 is a factor of $ab + 1$.
3. Create and print a list of integers from 50 to 100 (inclusive).
4. From the list in Exercise 3, create a sublist consisting of (a) even integers, (b) odd integers, (c) primes, and (d) numbers divisible by 13.
5. Create the two sets $A = \{1, 2, 3, 4, 5, 6\}$, $B = \{2, 4, 6, 8, 10\}$ and find (a) their intersection, (b) their union, and (c) the cardinality of their cartesian product.
6. Define a function `sum1ToN` that returns $1+2+3+\cdots+n$ using the formula $1+2+3+\cdots+n = \frac{n(n+1)}{2}$.
7. The function below prints string `obj` n times:

   ```
   def printNtimes(n,obj):
       count=0
       result=''      # empty string
       while count < n:
           result += str(obj)     # +=  is equivalent to result = result + str(obj)
           count += 1
       print(result)
   ```

   Test it with various inputs for `obj` and n.

   Notice there is no `return` line in the function. Since we don't ask the function to return anything we could either write the last line as `return`, or just leave it out as we have done above. Type in `print printNtimes(3,'hello')` and describe what happens.
8. Write a function `pow4(x)` that returns x^4 and performs only two multiplications (and no exponentiations).
9. Write a function `divBy(n,m)` which returns True if integer n is divisible by m, otherwise it returns False.

Appendix B

Basic Properties of Integers

In this book a few occasions have arisen where we needed some properties of the integers:

$$\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}.$$

This section is devoted to a brief account of some of these properties such as divisibility, greatest common divisors, the Euclidean Algorithm, and prime numbers.

B.1 Divisibility and the Euclidean Algorithm

We say that a **divides** b (written $a \mid b$) if and only if there is an integer d such that $b = ad$. For example, $2 \mid 6$, $12 \mid 60$, $-5 \mid 15$, and $8 \mid -24$. If a does not divide b then we write $a \nmid b$. For example $4 \nmid 2$ and $3 \nmid 4$.

We say d is the **greatest common divisor** of a and b (written $\gcd(a, b)$) if and only if

(i) $d \mid a$ and $d \mid b$, and

(ii) if $c \mid a$ and $c \mid b$ then $c \leq d$

Condition (i) says that d is a common divisor of a and b and condition (ii) says it is the greatest such divisor. For example, $\gcd(5, 15) = 5$ since 5 divides both 5 and 15 and it is the largest such divisor. Check for yourself that $\gcd(12, 32) = 4$.

If $\gcd(a, b) = 1$ then we say a and b are **relatively prime**. This is equivalent to saying a and b do not have a common prime factor, but we didn't phrase it in this way since we won't define what a prime number is until Section B.2. When a and b are small it is possible to see what $\gcd(a, b)$ is by inspection, but when a and b are large this is no longer possible. We will describe an algorithm for easily computing $\gcd(a, b)$ called the *Euclidean Algorithm* (see Theorem B.3). The Euclidean Algorithm allows us to compute gcd's without the need to factor the numbers first, and this is a good thing since factoring large numbers can be computationally difficult. Before we present the Euclidean Algorithm we present a useful theorem.

Theorem B.1 (Division Algorithm): *Let $a, b \in \mathbb{Z}$. Suppose that $b \neq 0$. Then there exist unique $q, r \in \mathbb{Z}$, with $0 \leq r < |b|$ such that*

$$a = qb + r.$$

For example, consider $a = 29$ and $b = 8$. Then $29 = 3 \cdot 8 + 5$. In this case $q = 3$ is called the *quotient* and $r = 5$ the *remainder*.

We will not give a formal proof of the Division Algorithm here, but the reader should be familiar with the method for finding q and r since it is just long division. The idea is to divide b evenly into a as many times as possible (this is q), and what is left over (the remainder r) must be smaller than b. For the interested reader a formal proof of this theorem is covered in any text in elementary number theory (for example [4]).

We can use Python commands $//$ and $\%$ to compute the quotient and remainder, respectively.

Sage

```
sage: 29//8   # this computes the quotient q
3
sage: 29%8  # this computes the remainder r
5
```

As an exercise in applying the definition of greatest common divisor, and because we will use this result in the proof of the Euclidean Algorithm, we prove the following lemma.

Lemma B.2: *If* $a = bq + r$ *then* $\gcd(a, b) = \gcd(b, r)$.

Proof: Let a, b, q and r be integers such that $a = bq + r$. Let $d = \gcd(a, b)$ and let $g = \gcd(b, r)$. We want to show $d = g$. Since $d \mid a$ and $d \mid b$ then there exist integers $\bar{a}$ and $\bar{b}$ such that $a = d\bar{a}$ and $b = d\bar{b}$. Substituting these into $a = bq + r$ we have

$$d\bar{a} = d\bar{b}q + r \implies r = d(\bar{a} - \bar{b}q)$$

which implies $d \mid r$. Therefore d divides both b and r so, by the definition of gcd, $d \leq g$.

On the other hand, since $g \mid b$ and $g \mid r$ then g divides $bq + r = a$ by a similar argument to the one we gave above. Therefore g divides both a and b so, by the definition of gcd, $g \leq d$.

It follows from $d \leq g$ and $g \leq d$ that $d = g$. □

Now we present the Euclidean Algorithm for computing greatest common divisors.

Theorem B.3 (Euclidean Algorithm): *If* a *and* b *are positive integers,* $b \neq 0$*, and*

$$\begin{aligned}
a &= qb + r, \quad 0 \leq r < b, \\
b &= q_1 r + r_1, \quad 0 \leq r_1 < r, \\
r &= q_2 r_1 + r_2, \quad 0 \leq r_2 < r_1, \\
&\vdots \qquad\qquad \vdots \\
r_k &= q_{k+2} r_{k+1} + r_{k+2}, \quad 0 \leq r_{k+2} < r_{k+1},
\end{aligned}$$

then for k *large enough, say* $k = \ell$*, we have* $r_{\ell+1} = 0$ *and*

$$r_{\ell-1} = q_{\ell+1} r_\ell,$$

and $\gcd(a, b) = r_\ell$.

Proof: Apply the Division Algorithm to a, b to produce q and r. Then apply the division algorithm to b, r to produce q_1 and r_1. If we continue applying the Division Algorithm, as indicated by each line in the statement of the theorem above, then we produce a strictly decreasing sequence of non-negative integers

$$b > r > r_1 > r_2 > \cdots$$

and this sequence must must come to an end. One of the remainders must be zero. Suppose that $r_{\ell+1} = 0$. Then $r_{\ell-1} = r_\ell q_{\ell+1}$. By Lemma B.2 applied over and over,

$$\gcd(a, b) = \gcd(b, r) = \gcd(r, r_1) = \gcd(r_1, r_2) = \cdots = \gcd(r_{\ell-1}, r_\ell) = \gcd(q_{\ell+1} r_\ell, r_\ell) = r_\ell.$$

□

If either a or b is negative we can use the fact that $\gcd(a, b) = \gcd(|a|, |b|)$.

Example B.1: Calculate $\gcd(343, 273)$.

Applying the Euclidean Algorithm

$$343 = 1 \cdot 273 + 70 \tag{B.1}$$
$$273 = 3 \cdot 70 + 63 \tag{B.2}$$
$$70 = 1 \cdot 63 + 7 \tag{B.3}$$
$$63 = 9 \cdot 7 \tag{B.4}$$

it follows that $\gcd(343, 273) = 7$.

The gcd command is implemented in Sage.

Sage

```
sage: gcd(343,273)
7
```

Let's continue with the previous example. From Equation B.3 we have

$$7 = 70 - 1 \cdot 63.$$

Solve Equation B.2 for 63 and plug into the previous equation

$$\begin{aligned} 7 &= 70 - 1(273 - 3 \cdot 70)) \\ &= 4 \cdot 70 - 1 \cdot 273. \end{aligned}$$

Now solve Equation B.1 for 70 and plug into the previous equation

$$\begin{aligned} 7 &= 4(343 - 1 \cdot 273) - 1 \cdot 273 \\ &= 4 \cdot 343 - 5 \cdot 273. \end{aligned}$$

The last equation shows that we are able to write the gcd of 343 and 273 as a linear combination of these two numbers: $7 = \gcd(343, 273) = 4 \cdot 343 - 5 \cdot 273$. This is not a coincidence as the next theorem states.

Theorem B.4 (Extended Euclidean Algorithm): *If $\gcd(a, b) = d$ then there exist integers u and v such that*

$$au + bv = d.$$

We won't present a detailed proof here, instead we will sketch the idea based on our work in Example B.1. The idea was to essentially work backwards through the Euclidean Algorithm. Beginning with Equation B.3 we solve for the gcd $r_2 = 7$ in terms of $r = 70$ and $r_1 = 63$. Keeping the gcd 7 on the left hand side of the equation we then we use the previous equations, B.2 and B.1, to first replace r_1 in terms of r and b, and then replace r in terms of a and b. This results in an equation for r in terms of a and b alone as desired.

The Extended Euclidean Algorithm is implemented in Sage. The command is `xgcd(a,b)` and it returns a triple (d, u, v) such that $d = \gcd(a, b)$ and $d = ua + vb$.

Sage

```
sage: print xgcd(343,273)
(7, 4, -5)
```

B.2 Prime Numbers

A **prime** is an integer that is greater than 1 and has no positive divisors other than 1 and itself. An integer that is greater than 1 but is not prime is called **composite**. For example 2, 3, and 5 are primes, but 4 and 6 are composite. By convention 1 is neither prime nor composite, we call it a **unit**. Thus the set of positive integers can be divided into three classes: the primes, the composites, and a unit.

There are infinitely many prime numbers and every integer n can be factored uniquely into a product of primes. Any text in elementary number theory will begin with proofs of these two statements (for example see [4]). It should be noted that one of the reasons we don't consider 1 to be prime is precisely because we want integers to have *unique* factorizations.

There is a lot we could say about prime numbers and all the properties they possess. However we will limit ourselves to stating one property, which could also be taken as the defining property for an integer to be prime.

Lemma B.5: *If p is a prime number and a and b are integers such that $p \mid ab$ then either $p \mid a$ or $p \mid b$.*

Proof: If $p \nmid a$ then $\gcd(p, a) = 1$ so by the Extended Euclidean Algorithm there exist integers u and v such that $1 = up + va$. Multiplying through by b we have

$$b = ubp + vab$$

and since p divides both terms on the right-hand side it must also divide b (see Exercise 2). □

Note that the statement of the lemma does not hold if p is not prime. See Exercise 11.

The study of integers and prime numbers is an active area of research today and there is still much about them that is unknown. For the interested reader a good place to start for further reading would be any text in elementary number theory (for example [4]).

B.3 Euler's ϕ-function

There is an important number-theoretic function, called *Euler's ϕ-function*, denoted by ϕ.

Definition B.1 (Euler's ϕ-Function): For any positive integer n, $\phi(n)$ is the number of integers in $\{1, 2, \dots, n\}$ which are relatively prime to n. In other words,

$$\phi(n) = |\{m \in \mathbb{Z} \mid 1 \leq m \leq n, \gcd(m, n) = 1\}|.$$

For example, $\phi(4) = 2$ since there are only two numbers in $\{1, 2, 3, 4\}$ that are relatively prime to 4, namely 1 and 3. $\phi(10) = 4$ since $1, 3, 7, 9$ are the only positive integers less than 10 which are relatively prime to 10. Notice that $\phi(p) = p - 1$ for any prime p since all the numbers $1, 2, 3, \dots, p - 1$ are relatively prime to p.

The following theorem gives a formula for $\phi(n)$ based on the prime factorization of n.

Theorem B.6: *If n has prime factorization given by*

$$n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k},$$

then

$$\phi(n) = p_1^{e_1-1}(p_1 - 1)p_2^{e_2-1}(p_2 - 1) \cdots p_k^{e_k-1}(p_k - 1).$$

For example, since $12 = 2^2 3$ then $\phi(12) = 2^{2-1}(2-1)3^{1-1}(3-1) = 2 \cdot 2 = 4$.

The reader can find a proof of Theorem B.6 in [4].

This function has been implemented in Sage, under the command `euler_phi()`. For example, here we see $\phi(96) = 32$.

Sage

```
sage: euler_phi(96)
32
```

To find the 32 numbers relative prime to 96 we can do the following.

Sage

```
sage: [ n for n in range(1,97) if gcd(n,96)==1 ]
[1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 55,
59, 61, 65, 67, 71, 73, 77, 79, 83, 85, 89, 91, 95]
```

Or we could use a filter with a lambda function.

Sage

```
sage: filter(lambda x: gcd(x,96)==1,range(1,97))
[1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49, 53, 55,
59, 61, 65, 67, 71, 73, 77, 79, 83, 85, 89, 91, 95]
```

B.4 Exercises

1. Which integers divide 0?

2. Prove that if $d \mid a$ and $d \mid b$ then $d \mid (a+b)$.

3. Show that if $a \mid b$ and $b \mid c$ then $a \mid c$.

4. If $d \mid c$ then prove that $d \mid ac$ for any integer a.

5. For each of the following pairs of numbers a, b below determine $\gcd(a,b)$. In each case use the Extended Euclidean Algorithm to find integers u and v such that $\gcd(a,b) = ua + bv$.

   (a) $306, 702$ (b) $314, 159$ (c) $4144, 7696$ (d) $888, 3071$

6. Find x and y such that $888x + 408y = 24$.

7. Find two different solutions of $299x + 247y = 13$.

8. What is $\gcd(n,1)$ where n is any positive integer?

9. What is $\gcd(n,0)$ where n is any positive integer?

10. If $\gcd(a,b) = d$ then show $\gcd(a/d, b/d) = 1$.

11. If $d \mid ab$ does it follow that $d \mid a$ or $d \mid b$?

12. Let $\gcd(a,b) = d$ and suppose that $c \mid a$ and $c \mid b$ show that $c \mid d$.

13. If $d \mid ab$ and $\gcd(d,a) = 1$ then show that $d \mid b$.
    (Hint: use the Extended Euclidean Algorithm.)

14. Calculate $\phi(42)$, $\phi(420)$, and $\phi(4200)$.

15. Prove that for $n \geq 1$, $\gcd(n-1, n) = 1$.

16. Find four solutions to $\phi(n) = 16$.

17. Prove this alternate formula for Euler's ϕ-function: if $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$, then

$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_k}\right).$$

    (Hint: Begin by using Theorem B.6.)

Bibliography

[1] A. F. Archer. A modern treatment of the 15-puzzle. *American Mathematical Monthly*, 106(9):793–799, 1999.

[2] C. Bandelow. *Inside Rubik's Cube and Beyond*. Birkhäuser, Boston, 1982.

[3] K. Conrad. The 15-puzzle (and rubik's cube). notes, 2008.

[4] U. Dudley. *Elementary Number Theory*. Dover Publications, 2nd edition, 2008.

[5] A. H. Frey Jr. and D. Singmaster. *Handbook of Cubik Math*. Enslow Publishers, New Jersey, 1982.

[6] J. A. Gallian. *Contemporary Abstract Algebra*. DC Heath and Company, Lexington, 1994.

[7] D. Joyner. *Adventures in Group Theory*. The John Hopkins University Press, Baltimore, 2nd edition, 2008.

[8] J.O. Kiltinen. How few transpositions suffice? ... you already know! *Mathematics Magazine*, 67(1):45–47, 1994.

[9] J.O. Kiltinen. *Oval Track and Other Permutation Puzzles: And Just Enough Group Theory to Solve Them*. The Mathematical Association of America, New York, 2003.

[10] J. Mulholland. Coures website for Permutation Puzzles: A Mathematical Perspective. `http://www.sfu.ca/∼jtmulhol/math302/`, 2012.

[11] J.G. Nourse. *The Simple Solution to Rubik's Cube*. Bantam Books, Toronto, 1981.

[12] N. Reilly. Applied algebraic systems. course notes, 1998.

[13] J. Scherphuis. Jaap's puzzle page. `http://www.jaapsch.net/puzzles/`, 2011.

[14] J. Slocum, D. Singmaster, D. Gebhardt, W. H. Huang, and G. Hellings. *The Cube: The Ultimate Guide to the World's Best Selling Puzzle*. Black Dog & Leventhal Publishers, Inc., New York, 2009.

[15] J. Slocum and D Sonneveld. *The 15-Puzzle: How it Drove the World Crazy*. The Slocum Puzzle Foundation, Beverly Hills, CA, 2006.

[16] W. A. Stein et al. *Sage*. The Sage Development Team, 2012. `http://www.sagemath.org`.

Index