Οι διαφάνειες βασίζονται σε αυτές του ακόλουθου μαθήματος:

Introduction to Algorithms (6-046J), MIT

https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/

Πόσο γρήγορα μπορούμε να ταξινομήσουμε;

Όλοι οι αλγόριθμοι ταξινόμησης που έχουμε δει είναι αλγόριθμοι ***συγκριτικής ταξινόμησης***: χρησιμοποιούν μόνο συγκρίσεις για να καθορίζουν τη σχετική διάταξη των στοιχείων.

- *Π.χ.*, ενθετική ταξινόμηση, συγχωνευτική ταξινόμηση, ταχυταξινόμηση, ταξινόμηση σωρού.

Ο καλύτερος χρόνος χειρότερης περίπτωσης που έχουμε δει για συγκριτική ταξινόμηση είναι $O(n \lg n)$.

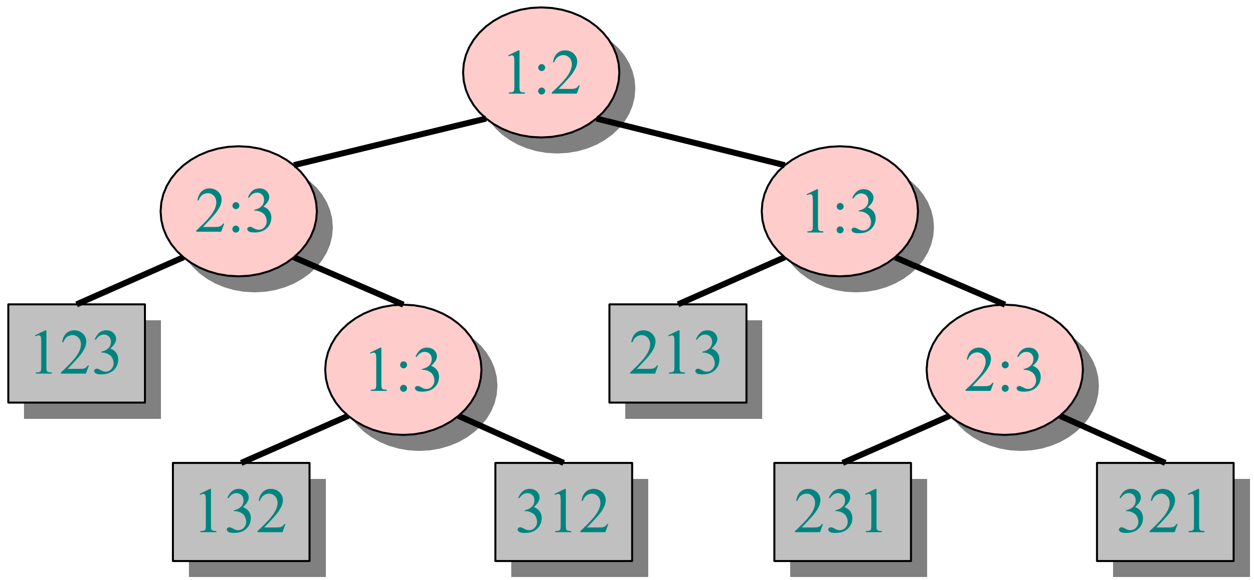
Είναι ο χρόνος $O(n\lg n)$ το καλύτερο που μπορούμε να κάνουμε;

Τα δένδρα μπορούν να βοηθήσουν στην απάντηση αυτής της ερώτησης.

Παράδειγμα Δένδρου Απόφασης

Ταξινόμηση της

$\langle a_1, a_2, \ldots, a_n \rangle$

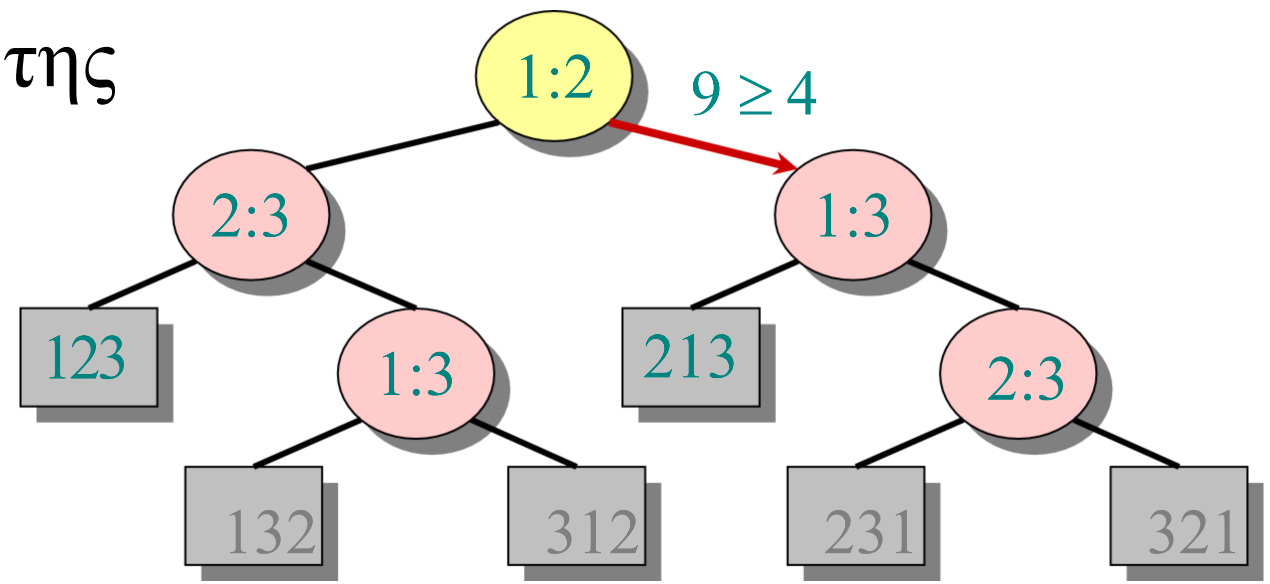


Κάθε εσωτερικός κόμβος επιγράφεται με $i{:}j$ για $i, j \in \{1, 2, \ldots, n\}$.

- Το αριστερό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \leq a_j$.
- Το δεξιό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \geq a_j$.

Παράδειγμα Δένδρου Απόφασης

Ταξινόμηση της $\langle a_1, a_2, a_3 \rangle = \langle 9, 4, 6 \rangle$

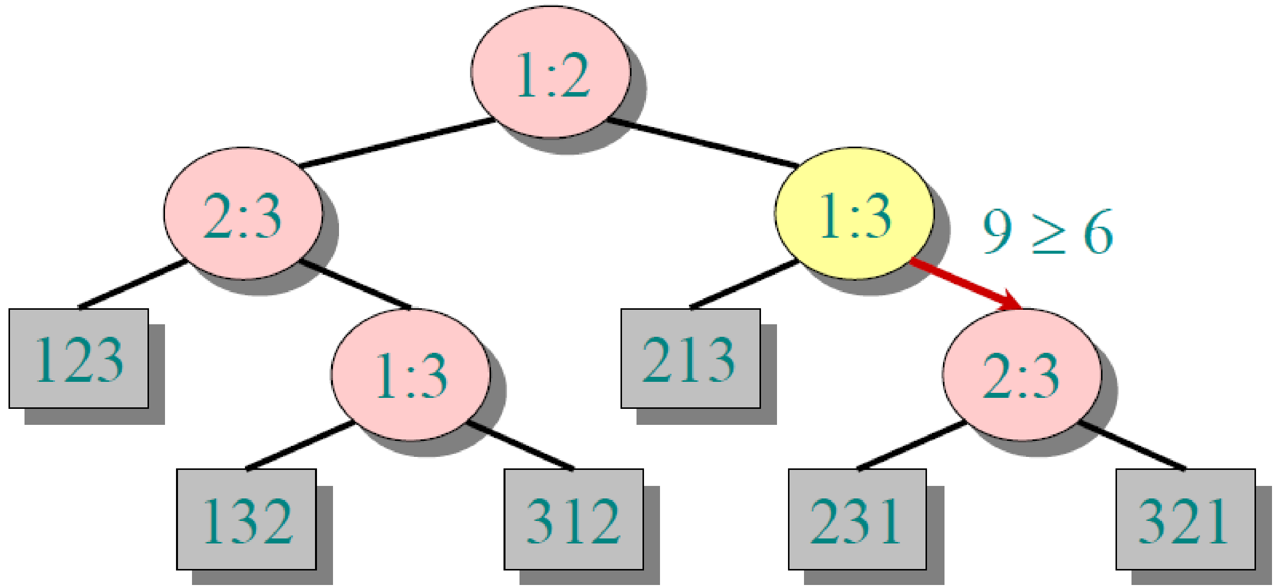


Κάθε εσωτερικός κόμβος επιγράφεται με $i{:}j$ για $i, j \in \{1, 2, \ldots, n\}$.

- Το αριστερό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \le a_j$.
- Το δεξιό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \ge a_j$.

Παράδειγμα Δένδρου Απόφασης

Ταξινόμηση της $\langle a_1, a_2, a_3\rangle = \langle\, 9, 4, 6\,\rangle$



Κάθε εσωτερικός κόμβος επιγράφεται με $i{:}j$ για $i, j \in \{1, 2,\ldots, n\}$.

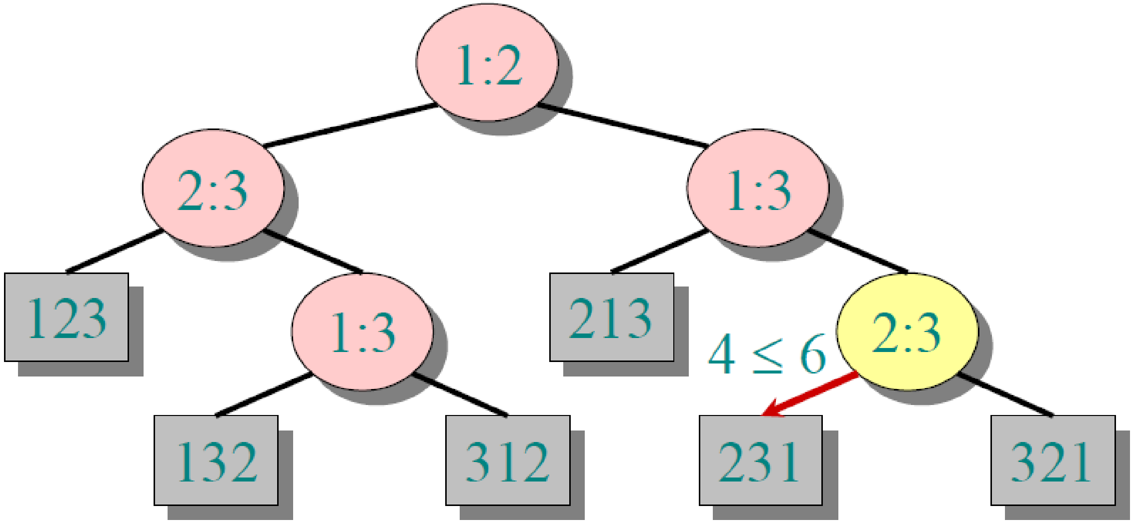
- Το αριστερό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \le a_j$.
- Το δεξιό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \ge a_j$.

Παράδειγμα Δένδρου Απόφασης

Ταξινόμηση της

$\langle a_1, a_2, a_3 \rangle$

$= \langle 9, 4, 6 \rangle$

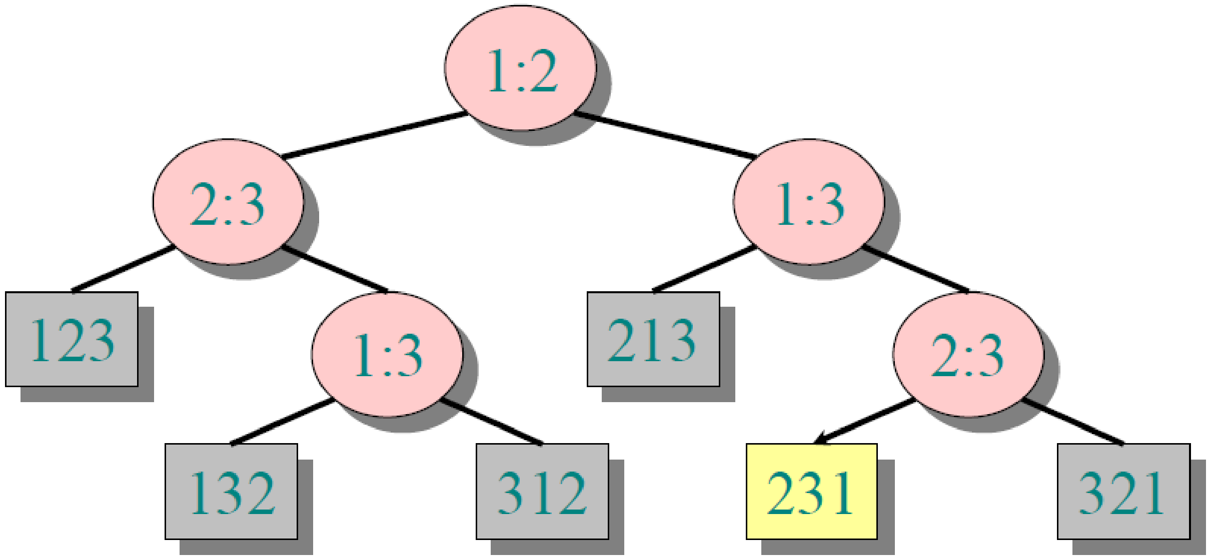


Κάθε εσωτερικός κόμβος επιγράφεται με $i{:}j$ για $i, j \in \{1, 2, \ldots, n\}$.

- Το αριστερό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \leq a_j$.
- Το δεξιό υποδένδρο δείχνει τις συγκρίσεις που έπονται αν $a_i \geq a_j$.

Παράδειγμα Δένδρου Απόφασης

Ταξινόμηση της $\langle a_1, a_2, a_3 \rangle = \langle\ 9, 4, 6\ \rangle$



$4 \leq 6 \leq 9$

Κάθε φύλλο περιέχει μία μετάθεση $\langle \pi(1), \pi(2), \ldots, \pi(n) \rangle$ η οποία δείχνει ότι η διάταξη $a_{\pi(1)} \leq a_{\pi(2)} \leq \ldots \leq a_{\pi(n)}$ έχει επιτευχθεί.

Μοντέλο Δένδρου Απόφασης

Ένα δένδρο απόφασης μπορεί να αποδώσει την εκτέλεση οποιασδήποτε συγκριτικής ταξινόμησης:

- Ένα δένδρο για κάθε μέγεθος εισόδου n.
- Δες τον αλγόριθμο ως διαχωριστή, οποτεδήποτε αυτός συγκρίνει δύο στοιχεία.
- Το δένδρο περιέχει τις συγκρίσεις για όλες τις πιθανές ροές ελέγχου.
- Ο χρόνος εκτέλεσης του αλγορίθμου = το μήκος του μονοπατιού που ακολουθείται.
- Χρόνος εκτέλεσης χειρότερης περίπτωσης = ύψος του δένδρου.

Κάτω όριο για Ταξινόμηση με βάση το Δένδρο Απόφασης

Θεώρημα. Κάθε δέντρο απόφασης που μπορεί να ταξινομήσει n στοιχεία πρέπει να έχει ύψος $\Omega(n \lg n)$.

Απόδειξη. Το δένδρο πρέπει να περιέχει $\geq n!$ φύλλα, αφού υπάρχουν $n!$ πιθανές μεταθέσεις. Ένα δυαδικό δένδρο ύψους h έχει $\leq 2^h$ φύλλα. Έτσι, $n! \leq 2^h$.

$$\begin{aligned} \therefore\ h &\geq \lg(n!) \\ &\geq \lg((n/e)^n) \\ &= n \lg n - n \lg e \\ &= \Omega(n \lg n) \end{aligned}$$

(lg είναι μονότονη αύξουσα συνάρτηση)

(προσέγγιση Stirling)

Κάτω όριο για Ταξινόμηση με βάση το Δένδρο Απόφασης

Πόρισμα. Η ταξινόμηση σωρού και η συγχωνευτική ταξινόμηση είναι ασυμπτωτικά βέλτιστοι αλγόριθμοι συγκριτικής ταξινόμησης.

Ταξινόμηση σε γραμμικό χρόνο

Απαριθμητική ταξινόμηση: Όχι συγκρίσεις μεταξύ των στοιχείων.

- ***Είσοδος***: $A[1\,.\,.\,n]$, όπου $A[j] \in \{1, 2, \ldots, k\}$.
- ***Έξοδος***: $B[1\,.\,.\,n]$, ταξινομημένος.
- ***Βοηθητικός πίνακας***: $C[1\,.\,.\,k]$.

Απαριθμητική ταξινόμηση

for $i \leftarrow 1$ **to** k
  do $C[i] \leftarrow 0$
for $j \leftarrow 1$ **to** n
  do $C[A[j]] \leftarrow C[A[j]] + 1$ $\triangleright\ C[i] = |\{\text{key} = i\}|$
for $i \leftarrow 2$ **to** k
  do $C[i] \leftarrow C[i] + C[i-1]$ $\triangleright\ C[i] = |\{\text{key} \leq i\}|$
for $j \leftarrow n$ **downto** 1
  do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$
    $C[A[j]] \leftarrow C[A[j]] - 1$

Παράδειγμα Απαριθμητικής Ταξινόμησης

	1	2	3	4	5
A:	4	1	3	4	3

B:					

	1	2	3	4
C:				

Βρόχος 1

	1	2	3	4	5
A:	4	1	3	4	3

	1	2	3	4
C:	0	0	0	0

B:					

for $i \leftarrow 1$ **to** k

  do $C[i] \leftarrow 0$

Βρόχος 2

	1	2	3	4	5
A:	4	1	3	4	3

	1	2	3	4
C:	0	0	0	1

B:					

for $j \leftarrow 1$ **to** n

do $C[A[j]] \leftarrow C[A[j]] + 1 \quad \triangleright\ C[i] = |\{\text{key} = i\}|$

Βρόχος 2

A:

1	2	3	4	5
4	1	3	4	3

C:

1	2	3	4
1	0	0	1

B:

for $j \leftarrow 1$ **to** n
  do $C[A[j]] \leftarrow C[A[j]] + 1$ $\quad \triangleright\ C[i] = |\{\text{key} = i\}|$

Βρόχος 2

A:

1	2	3	4	5
4	1	3	4	3

C:

1	2	3	4
1	0	1	1

B:

for $j \leftarrow 1$ **to** n

  do $C[A[j]] \leftarrow C[A[j]] + 1$ $\triangleright\ C[i] = |\{\text{key} = i\}|$

Βρόχος 2

	1	2	3	4	5
A:	4	1	3	4	3

	1	2	3	4
C:	1	0	1	2

B:					

for $j \leftarrow 1$ **to** n

  do $C[A[j]] \leftarrow C[A[j]] + 1$ $\triangleright\ C[i] = |\{\text{key} = i\}|$

Βρόχος 2

A:

1	2	3	4	5
4	1	3	4	3

C:

1	2	3	4
1	0	2	2

B:

for $j \leftarrow 1$ **to** n

  do $C[A[j]] \leftarrow C[A[j]] + 1 \quad \triangleright\ C[i] = |\{\text{key} = i\}|$

Βρόχος 3

	1	2	3	4	5
A:	4	1	3	4	3
B:					

	1	2	3	4
C:	1	0	2	2
C':	1	1	2	2

for $i \leftarrow 2$ **to** k

  do $C[i] \leftarrow C[i] + C[i-1]$ $\triangleright\ C[i] = |\{\text{key} \leq i\}|$

Βρόχος 3

	1	2	3	4	5
A:	4	1	3	4	3
B:					

	1	2	3	4
C:	1	0	2	2
C':	1	1	3	2

for $i \leftarrow 2$ **to** k

do $C[i] \leftarrow C[i] + C[i-1]$ $\qquad \triangleright\ C[i] = |\{\text{key} \leq i\}|$

Βρόχος 3

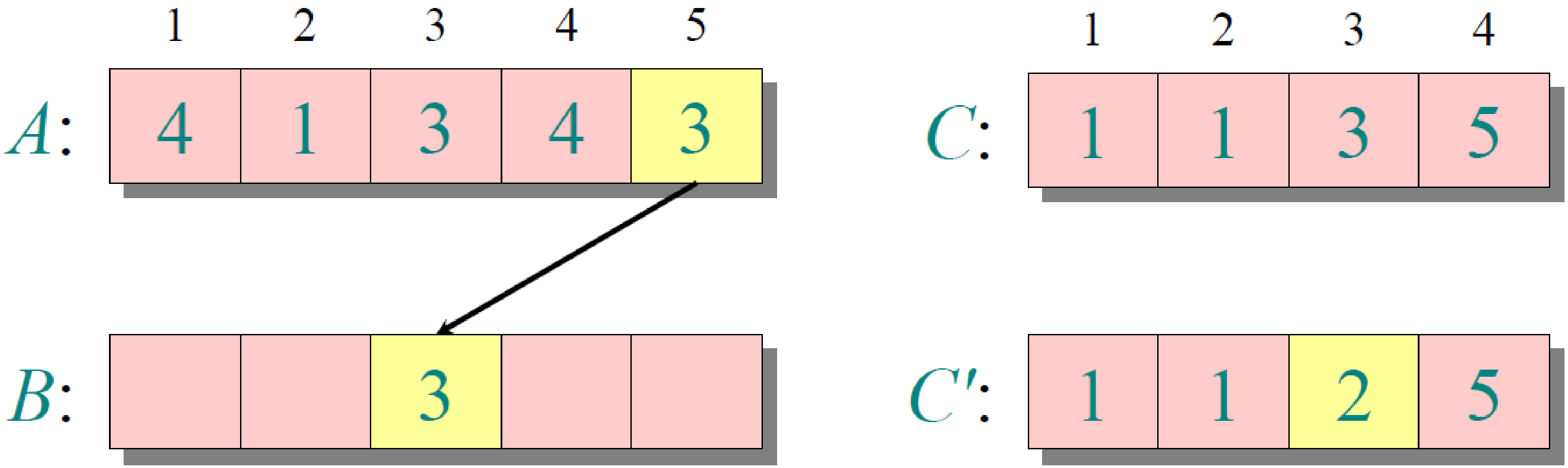
	1	2	3	4	5
A:	4	1	3	4	3
B:					

	1	2	3	4
C:	1	0	2	2
C':	1	1	3	5

for $i \leftarrow 2$ **to** k

  do $C[i] \leftarrow C[i] + C[i-1]$ $\triangleright$ $C[i] = |\{\text{key} \leq i\}|$

Βρόχος 4

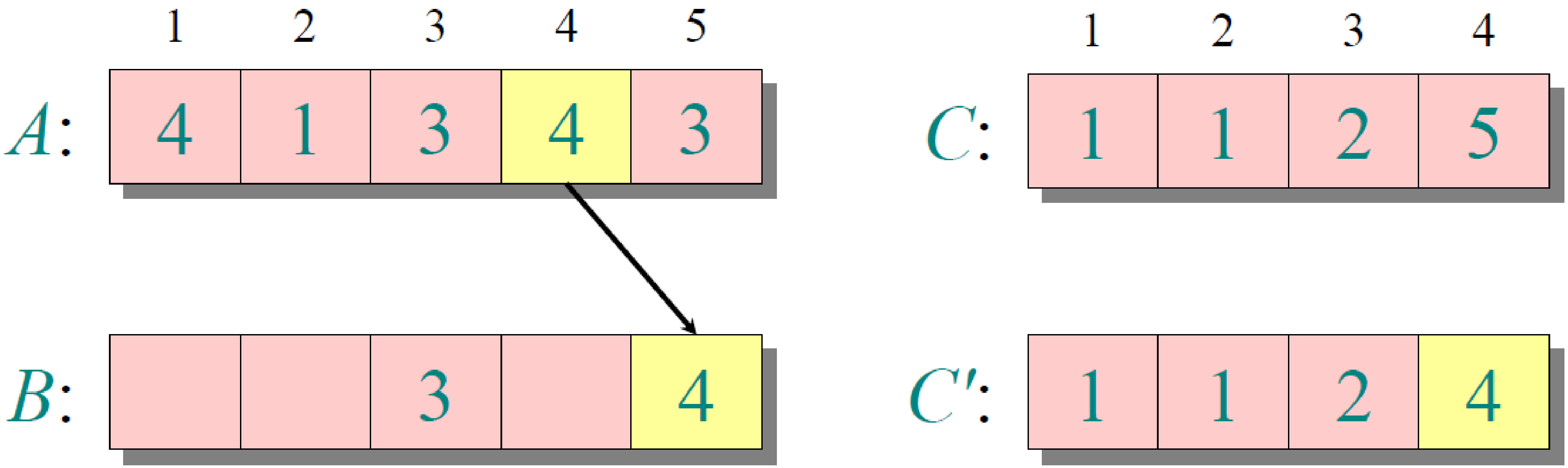


for $j \leftarrow n$ **downto** 1

  do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$

    $C[A[j]] \leftarrow C[A[j]] - 1$

Βρόχος 4

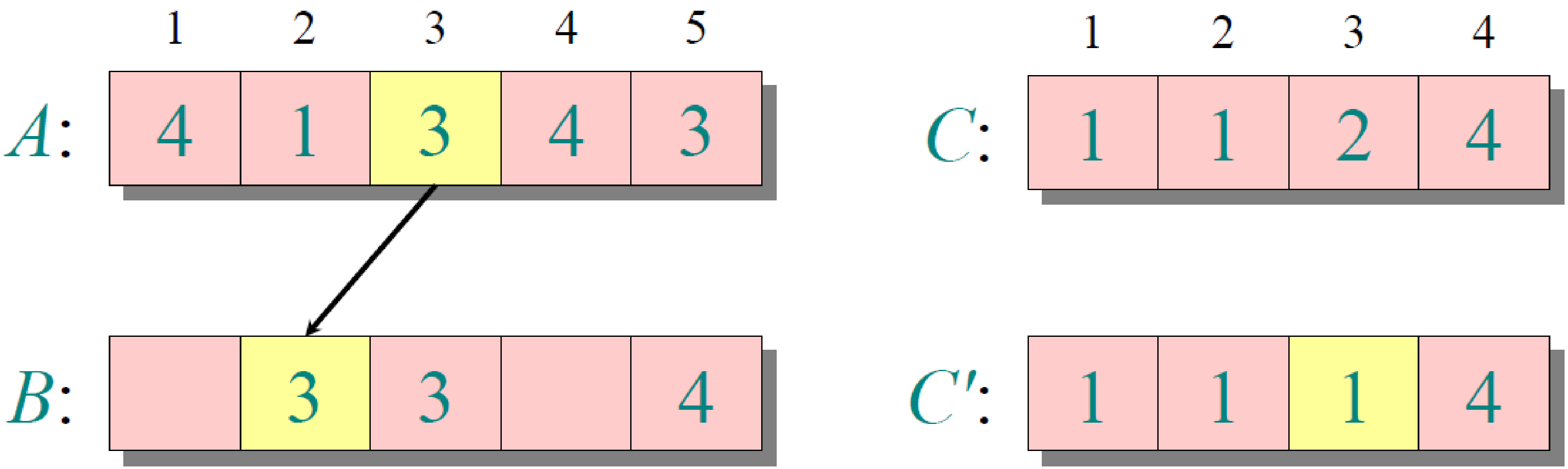


for $j \leftarrow n$ **downto** 1

  do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$

    $C[A[j]] \leftarrow C[A[j]] - 1$

Βρόχος 4

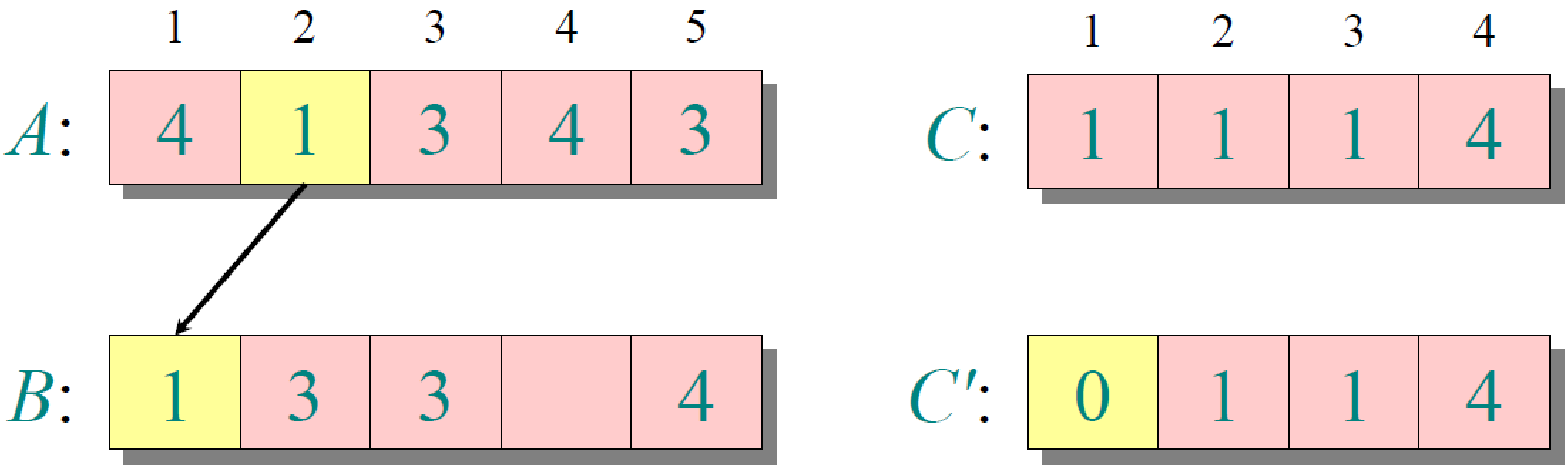


for $j \leftarrow n$ **downto** 1

do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$

$C[A[j]] \leftarrow C[A[j]] - 1$

Βρόχος 4

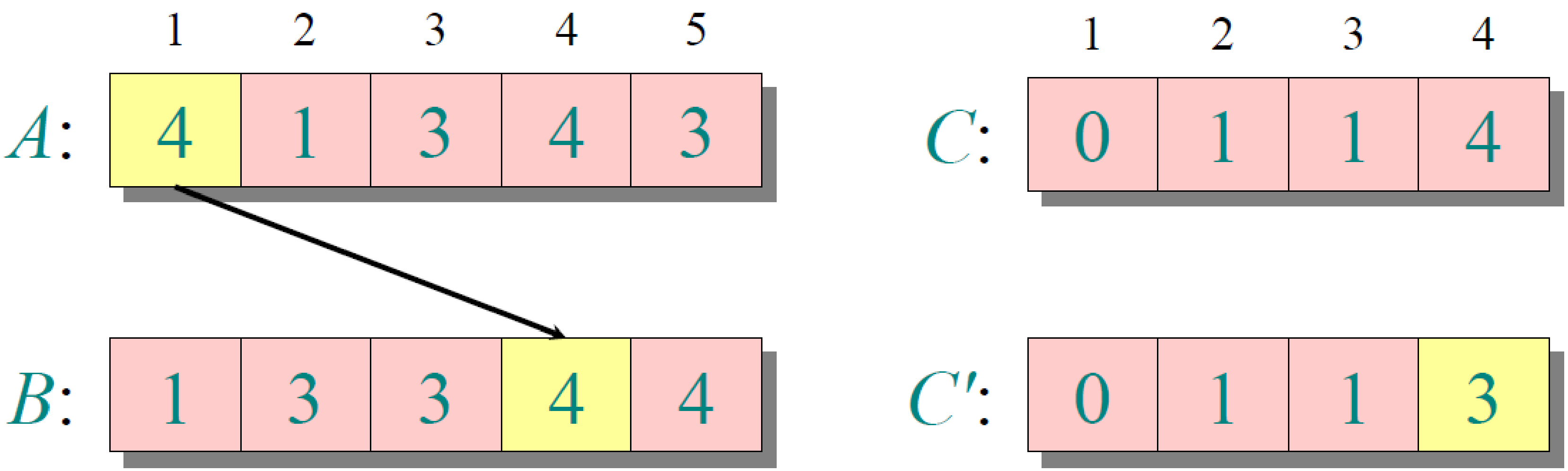


for $j \leftarrow n$ **downto** 1

do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$

$C[A[j]] \leftarrow C[A[j]] - 1$

Loop 4



for $j \leftarrow n$ **downto** 1

  do $B[C[A[j]]] \leftarrow \mathrm{A}[j]$

    $C[A[j]] \leftarrow C[A[j]] - 1$

Ανάλυση

$$\Theta(k) \left\{ \begin{array}{l} \textbf{for } i \leftarrow 1 \textbf{ to } k \\ \quad \textbf{do } C[i] \leftarrow 0 \end{array} \right.$$

$$\Theta(n) \left\{ \begin{array}{l} \textbf{for } j \leftarrow 1 \textbf{ to } n \\ \quad \textbf{do } C[A[j]] \leftarrow C[A[j]] + 1 \end{array} \right.$$

$$\Theta(k) \left\{ \begin{array}{l} \textbf{for } i \leftarrow 2 \textbf{ to } k \\ \quad \textbf{do } C[i] \leftarrow C[i] + C[i-1] \end{array} \right.$$

$$\Theta(n) \left\{ \begin{array}{l} \textbf{for } j \leftarrow n \textbf{ downto } 1 \\ \quad \textbf{do } B[C[A[j]]] \leftarrow \mathrm{A}[j] \\ \qquad C[A[j]] \leftarrow C[A[j]] - 1 \end{array} \right.$$

$$\Theta(n + k)$$

Χρόνος Εκτέλεσης

Αν $k = O(n)$, τότε η απαριθμητική ταξινόμηση χρειάζεται χρόνο $\Theta(n)$.

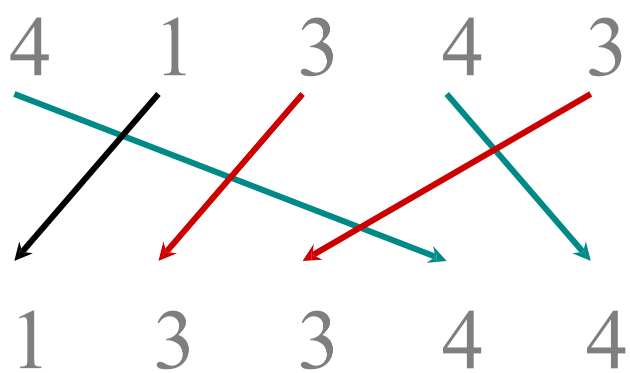
- Αλλά, η ταξινόμηση χρειάζεται $\Omega(n \lg n)$ χρόνο!
- Που είναι η πλάνη;

Απάντηση:

- Η ***συγκριτική ταξινόμηση*** παίρνει χρόνο $\Omega(n \lg n)$.
- Η απαριθμητική ταξινόμηση δεν είναι ***συγκριτική ταξινόμηση***.
- Στην πραγματικότητα, δεν γίνεται καμία σύγκριση μεταξύ των στοιχείων.

Ευσταθής ταξινόμηση

Η απαριθμητική ταξινόμηση είναι ευσταθής: διατηρεί την αρχική διάταξη των ίσων στοιχείων της εισόδου:



Αριθμοτακτική Ταξινόμηση (Radix Sort)

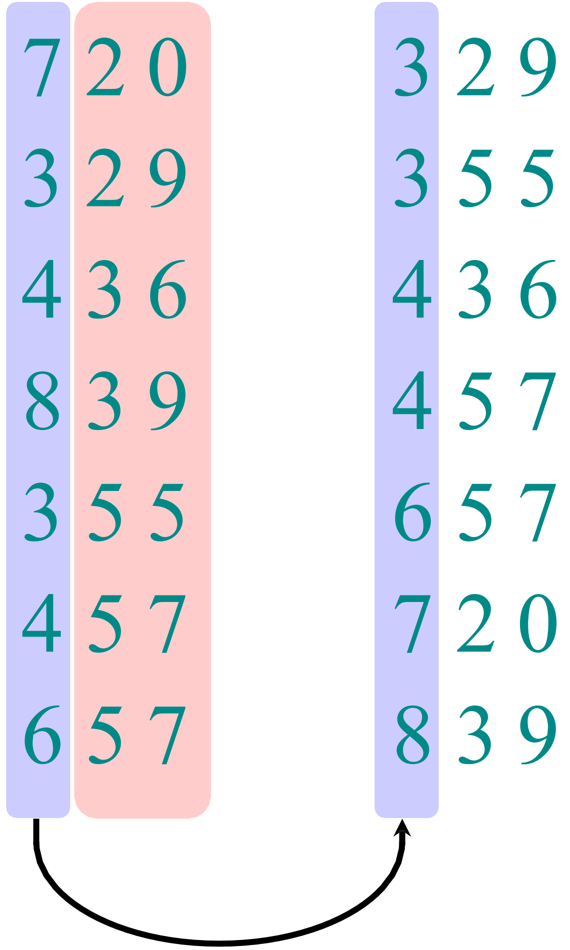
3 2 9	7 2 0	7 2 0	3 2 9
4 5 7	3 5 5	3 2 9	3 5 5
6 5 7	4 3 6	4 3 6	4 3 6
8 3 9	4 5 7	8 3 9	4 5 7
4 3 6	6 5 7	3 5 5	6 5 7
7 2 0	3 2 9	4 5 7	7 2 0
3 5 5	8 3 9	6 5 7	8 3 9

Ορθότητα της αριθμοτακτικής ταξινόμησης

Επαγωγή ως προς τη θέση του ψηφίου

Υποθέτουμε ότι οι αριθμοί είναι ταξινομημένοι στα χαμηλότερης τάξης $t-1$ ψηφία.

- Ταξινόμηση στο ψηφίο t

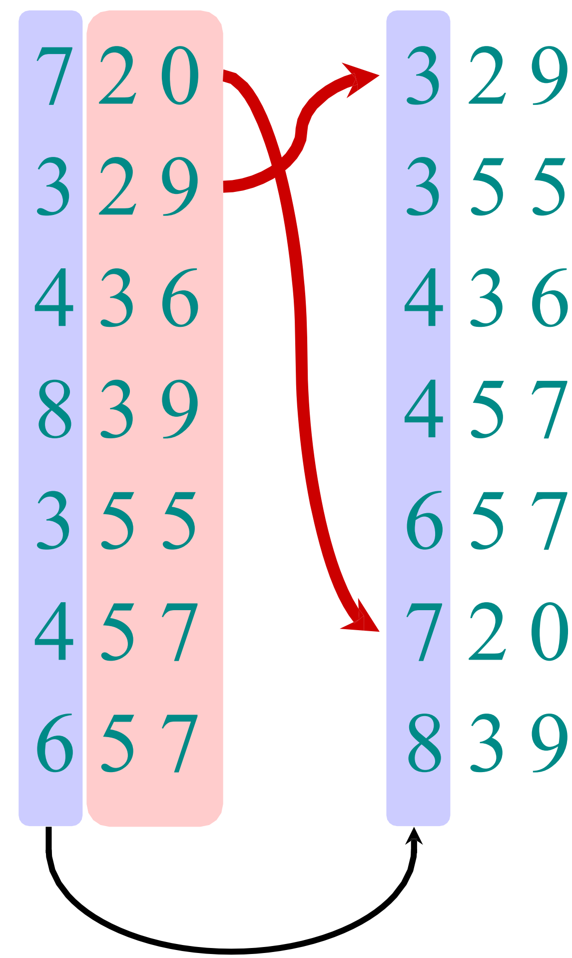


Ορθότητα της αριθμοτακτικής ταξινόμησης

Επαγωγή ως προς τη θέση του ψηφίου

Υποθέτουμε ότι οι αριθμοί είναι ταξινομημένοι στα χαμηλότερης τάξης $t-1$ ψηφία.

- Ταξινόμηση στο ψηφίο t
  - Δύο αριθμοί οι οποίοι διαφέρουν στο ψηφίο t ταξινομούνται σωστά.

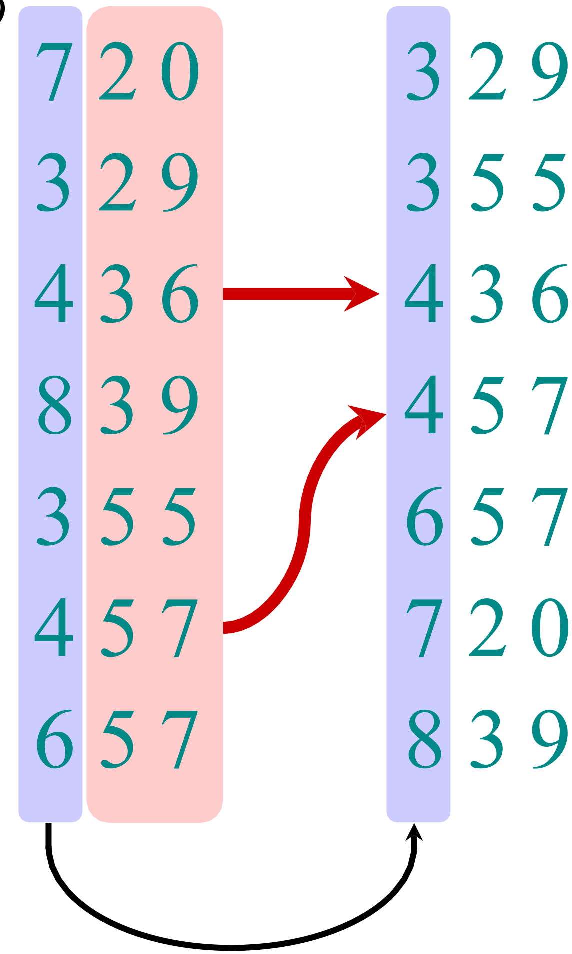


Ορθότητα της αριθμοτακτικής ταξινόμησης

Επαγωγή ως προς τη θέση του ψηφίου

Υποθέτουμε ότι οι αριθμοί είναι ταξινομημένοι στα χαμηλότερης τάξης $t-1$ ψηφία.

- Ταξινόμηση στο ψηφίο t
  - Δύο αριθμοί οι οποίοι διαφέρουν στο ψηφίο t ταξινομούνται σωστά.
  - Δύο αριθμοί ίδιοι στο ψηφίο t τοποθετούνται στην ίδια σειρά όπως στην είσοδο $\Rightarrow$ σωστή διάταξη.



Ανάλυση της αριθμοτακτικής ταξινόμησης

- Υποθέτουμε απαριθμητική ταξινόμηση σε κάθε πέρασμα.
- Ταξινόμησε n αριθμούς των b δυαδικών ψηφίων ο καθένας.

Παράδειγμα: 32-bit αριθμοί

8	8	8	8

- Κάθε αριθμός μπορεί να θεωρηθεί σαν να έχει b/r ψηφία βάσης 2^r

$r = 8 \Rightarrow b/r = 4$ περάσματα της απαριθμητικής ταξινόμησης σε ψηφία βάσης 2^8 ή $r = 16 \Rightarrow b/r = 2$ περάσματα απαριθμητικής ταξινόμησης σε ψηφία βάσης 2^{16}.

Πόσα περάσματα πρέπει να κάνουμε;

Ανάλυση (συν.)

Θυμηθείτε: Η απαριθμητική ταξινόμηση παίρνει χρόνο $\Theta(n + k)$ για να ταξινομήσει n αριθμούς στο εύρος $0 \ldots k - 1$.

Αν κάθε b-bit αριθμός σπάσει σε r-bit κομμάτια, κάθε πέρασμα της απαριθμητικής ταξινόμησης παίρνει χρόνο $\Theta(n + 2^r)$. Αφού υπάρχουν b/r περάσματα, έχουμε

$$T(n,b) = \Theta\left(\frac{b}{r}\left(n + 2^r\right)\right).$$

Διάλεξε το r για ελαχιστοποίηση του $T(n,b)$:

- Αυξάνοντας το r προκύπτουν λιγότερα περάσματα, αλλά καθώς $r \gg \lg n$, ο χρόνος αυξάνεται εκθετικά.

Επιλογή *r*

$$T(n,b) = \Theta\left(\frac{b}{r}\left(n + 2^r\right)\right).$$

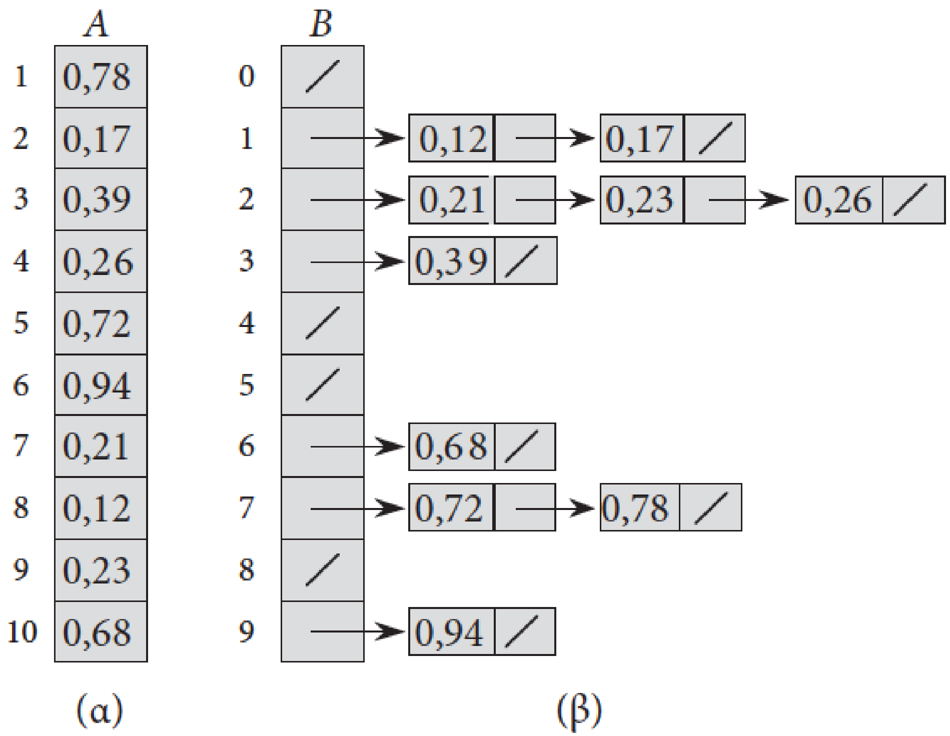
Ελαχιστοποίηση του $T(n, b)$ με παραγώγιση και εξίσωση με το 0.

Ή, απλά παρατηρούμε ότι δεν θέλουμε $2^r \gg n$, και ότι δεν υπάρχει ζημιά αν επιλέξουμε το r όσο το δυνατόν μεγαλύτερο χωρίς να παραβιάσουμε τον παραπάνω περιορισμό.

Επιλέγοντας $r = \lg n$ συνεπάγεται $T(n,b) = \Theta(bn/\lg n)$.

- Για αριθμούς στο εύρος $0 \ldots n^d - 1$, έχουμε $b = d \lg n$ $\Rightarrow$ η αριθμοτακτική ταξινόμηση τρέχει σε χρόνο $\Theta(d\, n)$

Ταξινόμηση με δοχεία



Σχήμα 8.4 Η λειτουργία της διαδικασίας Ταξινομηση με Δοχεια για $n = 10$. **(α)** Η συστοιχία εισόδου $A[1\,..\,10]$. **(β)** Η συστοιχία $B[0\,..\,9]$ των ταξινομημένων λιστών (δοχείων) μετά τη γραμμή 8 του αλγορίθμου. Το δοχείο υπ' αριθμ. i περιέχει τις τιμές που βρίσκονται στο ημιανοικτό διάστημα $[i/10, (i+1)/10)$. Η ταξινομημένη ακολουθία εξόδου συντίθεται με διαδοχική συναρμογή των λιστών $B[0], B[1], \ldots, B[9]$.

```
BUCKET-SORT(A)
let B[0..n − 1] be a new array
n = A.length
for i = 0 to n − 1
    make B[i] an empty list
for i = 1 to n
    insert A[i] into list B[⌊n A[i]⌋]
for i = 0 to n − 1
    sort list B[i] with insertion sort
concatenate the lists B[0], B[1], ..., B[n − 1] together in order
```

$$T(n) = \Theta(n) + \sum_{i=0}^{n-1} O(n_i^2)\ .$$

$$\begin{aligned}
\mathrm{E}\left[T(n)\right] &= \mathrm{E}\left[\Theta(n) + \sum_{i=0}^{n-1} O(n_i^2)\right] \\
&= \Theta(n) + \sum_{i=0}^{n-1} \mathrm{E}\left[O(n_i^2)\right] \\
&= \Theta(n) + \sum_{i=0}^{n-1} O\left(\mathrm{E}\left[n_i^2\right]\right)
\end{aligned}$$

$$\mathrm{E}\left[n_i^2\right] = 2 - 1/n$$

$$X_{ij} = \mathrm{I}\left\{\text{το } A[j] \text{ καταλήγει στο δοχείο } i\right\}$$

$$n_i = \sum_{j=1}^{n} X_{ij}$$

$$\begin{aligned}
\mathrm{E}\left[n_i^2\right] &= \mathrm{E}\left[\left(\sum_{j=1}^{n} X_{ij}\right)^2\right] \\
&= \mathrm{E}\left[\sum_{j=1}^{n}\sum_{k=1}^{n} X_{ij}X_{ik}\right] \\
&= \mathrm{E}\left[\sum_{j=1}^{n} X_{ij}^2 + \sum_{1\le j\le n}\ \sum_{\substack{1\le k\le n \\ k\ne j}} X_{ij}X_{ik}\right] \\
&= \sum_{j=1}^{n}\mathrm{E}\left[X_{ij}^2\right] + \sum_{1\le j\le n}\ \sum_{\substack{1\le k\le n \\ k\ne j}} \mathrm{E}\left[X_{ij}X_{ik}\right] ,
\end{aligned}$$

$$\begin{aligned} \mathrm{E}\left[X_{ij}^2\right] &= 1^2 \cdot \frac{1}{n} + 0^2 \cdot \left(1 - \frac{1}{n}\right) \\ &= \frac{1}{n} . \end{aligned}$$

Όταν k <> j, οι μεταβλητέςXij και Xik είναι ανεξάρτητες μεταξύ τους, και άρα

$$\begin{aligned} \mathrm{E}\left[X_{ij} X_{ik}\right] &= \mathrm{E}\left[X_{ij}\right] \mathrm{E}\left[X_{ik}\right] \\ &= \frac{1}{n} \cdot \frac{1}{n} \\ &= \frac{1}{n^2} . \end{aligned}$$

$$\begin{aligned} \mathrm{E}\left[n_i^2\right] &= \sum_{j=1}^{n} \frac{1}{n} + \sum_{1 \le j \le n} \sum_{\substack{1 \le k \le n \\ k \ne j}} \frac{1}{n^2} \\ &= n \cdot \frac{1}{n} + n(n-1) \cdot \frac{1}{n^2} \\ &= 1 + \frac{n-1}{n} \\ &= 2 - \frac{1}{n} , \end{aligned}$$

$$\boxed{\Theta(n) + n \cdot O(2 - 1/n) = \Theta(n)}$$