# Εισαγωγή στο Unix / Linux

*(μέρος γ')*

Παναγιώτης Κοτζανικολάου
pkotzani@unipi.gr

Κορέας Πλάτων
platonas@gmail.com

# Το περιβάλλον του χρήστη

- Το περιβάλλον (environment) του χρήστη είναι ένα σύνολο μεταβλητών και τιμών
- Μια μεταβλητή είναι ένα string (σειρά χαρακτήρων) που προβάλει βασικές πληροφορίες για το λειτουργικό σύστημα
- Η εντολή echo εμφανίζει στην οθόνη τα ορίσματα του σας χαρακτήρες

- Χρήση της εντολής:

  ***$ echo HOME***

# Κοινές μεταβλητές περιβάλλοντος

- Για την προβολή των μεταβλητών χρησιμοποιούμε την echo και προσθέτουμε στην αρχή της μεταβλητής το σύμβολο $

- HOME : η μεταβλητή αυτή εμφανίζει το πλήρες όνομα της διαδρομής του home καταλόγου. Ο κατάλογος αυτός είναι ο προκαθορισμένος κατά την σύνδεση του χρήστη στο σύστημα

- Χρήση της εντολής:

  ***$ echo $HOME***

# Κοινές μεταβλητές περιβάλλοντος

- LOGNAME: η μεταβλητή εμφανίζει το όνομα του λογαριασμού ή το userID του χρήστη
- MAIL: η μεταβλητή εμφανίζει το κατάλογο αποθήκευσης του γραμματοκιβωτίου του χρήστη
- SHELL: η μεταβλητή εμφανίζει το όνομα του shell που χρησιμοποιεί ο χρήστης
- ***$ echo $LOGNAME***
- ***$ echo $SHELL***

# Κοινές μεταβλητές περιβάλλοντος

- PATH: η μεταβλητή εμφανίζει τους καταλόγους στους οποίους πρέπει να αναζητήσουν τα προγράμματα για να μπορέσουν να εκτελέσουν τις εντολές τους. Η μεταβλητή αυτή μπορεί να τροποποιηθεί από τον διαχειριστή για να μπορεί να παραμετροποιήσει τον τρόπο που εκτελούνται τα προγράμματα στους χρήστες

- ***$ echo $PATH***

# Προγραμματισμός bash shell

- Προγραμματισμός κελύφους είναι η διαδικασία δημιουργίας σεναρίων κελύφους (shell scripts) ή προγραμμάτων κελύφους (shell programs)

- Ένα shell script είναι ένα αρχείο που περιέχει μία η περισσότερες εντολές

- Στην περίπτωση που το αρχείο shell δεν βρίσκεται στο PATH του χρήστη εισάγουμε το αρχείο ως “./scriptname” αλλιώς το εκτελούμε με το όνομα του

# Προγραμματισμός bash shell

- Τα Shell scripts ξεκινάνε με την εντολή #! και το όνομα του shell
  - Παράδειγμα : **#!/bin/sh** για το Bourne shell
  - Σε περίπτωση που δεν ενσωματώσουμε την παραπάνω εντολή γίνεται χρήση του shell του χρήστη
- Όλες οι εντολές Unix μπορούν να ενσωματωθούν σε ένα shell script
  - Οι εντολές εκτελούνται με την σειρά που θα τις θέσουμε στον κώδικα μας και με τις δομές ελέγχου που θα τους ορίσουμε
- Τα διαφορετικά shells έχουν και διαφορετικές δομές ελέγχου
  - Στα παραδείγματα που ακολουθούν χρησιμοποιούμε το Bourne shell (sh) κέλυφος

# Γιατί γράφουμε shell scripts ;

- για να αποφύγουμε μια επαναληπτική διαδικασία. Είναι ευκολότερο να δημιουργήσουμε ένα σενάριο το οποίο θα τρέχει τα πρότυπα των εντολών που επιθυμούμε με μία μόνο εντολή

- για την αυτοματοποίηση δύσκολων διαδικασιών οι οποίες περιλαμβάνουν πολλές εντολές με πολλά ορίσματα και είναι δύσκολο να τα καταλάβουμε ή να τα θυμόμαστε κάθε φορά που εκτελούμε αυτή την διαδικασία.

# Μεταβλητές

- Στην δήλωση των μεταβλητών σε ένα πρόγραμμα shell υπάρχει διάκριση μεταξύ πεζών και κεφαλαίων (case-sensitive ) πχ. foo, FOO, Foo

- Με την προσθήκη του **$** αποθηκεύουμε τις τιμές για τις μεταβλητές ενώ με την προσθήκη **#** σχολιάζουμε μια γραμμή

  ***$ salutation=Hello***

  ***$ echo $salutation***

  ***$ Hello***

# Χρήση των Quotes

- Έστω ένα αρχείο test με περιεχόμενο

```
#!/bin/sh
# Σχόλιο γραμμής
myvar="metabliti" #Δημιουργία μεταβλητής
echo $myvar   # Εκτύπωση μεταβλητής
echo "$myvar"  # Εκτύπωση μεταβλητής με διπλά "
echo '$myvar' # Εκτύπωση του μηνύματος της echo με μονά '
echo  Εισάγετε κείμενο
read myvar # Εκχώρηση νέας τιμής για την myvar
echo  Η μεταβλητή '$myvar' έχει την νέα τιμή $myvar
```

- Με την χρήση των διπλών quotes οι μεταβλητές μπορούν να εμφανίζουν τις τιμές εκχώρησης τους σε αντίθεση με τα μονά

# Λογικές εκφράσεις (boolean)

- Σχεσιακοί Τελεστές για αριθμητικές πράξεις:
  **-eq, -ne, -gt, -ge, -lt, -le**

- Τελεστές αρχείων:
  **-f file** Αληθής εάν το αρχείο υπάρχει και δεν είναι φάκελος
  **-d file** Αληθής εάν ο φάκελος υπάρχει
  **-s file** Αληθής εάν το αρχείο υπάρχει και έχει size > 0

- Τελεστές String :
  **-z string** Εάν το μήκος είναι 0
  **-n string** Εάν το μήκος δεν είναι 0

# Παράδειγμα shell script

- Δημιουργούμε ένα αρχείο **test** με vi ή nano

***#! /bin/bash***

***date***

***uname -a***

***whoami***

- Δίνουμε δικαιώματα εκτέλεσης για το αρχείο αυτό

***chmod 755 test***

- Εκτελούμε το αρχείο

***./ test***

# Shell script με μεταβλητή

- Δημιουργούμε και εκτελούμε το αρχείο **test1**

```
#!/bin/sh
number=1 # Ορισμός αριθμού
name= " John Doe " #Ορισμός string με διπλά quotes
echo $number $name
```

# Shell script με είσοδο δεδομένων

- Δημιουργούμε και εκτελούμε το αρχείο **test2**

```
#!/bin/sh
echo Enter  name:
read name #  Είσοδος δεδομένων
echo  Hello $name!
```

# Shell script με είσοδο εντολών

- Δημιουργούμε και εκτελούμε το αρχείο **test3**

```
#!/bin/sh
user=$(whoami) # Ορισμός της εντολής whoami
echo  Hi $user!

path=$(pwd) # Ορισμός της εντολής pwd
echo  To  $path!
```

# Shell script με την if

- Δημιουργούμε και εκτελούμε το αρχείο **test4**

```
#!/bin/sh
xristis="admin"
#Ελεγχος συνθήκης if
if [ $xristis = "admin" ] ; then
echo "Hi $xristis“
fi
```

# Shell script με την if και else

- Δημιουργούμε και εκτελούμε το αρχείο **test5**

```
#! /bin/sh
echo " Εισάγετε το όνομα του αρχείου που θέλετε βρείτε
   στον τρέχοντα φάκελο "
read filetype #Είσοδος δεδομένων
#Ελεγχος συνθήκης if  και else
if [ -f $filetype ] ; then
        echo " Υπάρχει το συγκεκριμένο αρχείο "
else
        echo "Δεν υπάρχει το συγκεκριμένο αρχείο"
fi
```

# Shell script με την if, elif και else

- Δημιουργούμε και εκτελούμε το αρχείο **test6**

```
#!/bin/bash
echo "Δώσε δυο ακεραίους "; read num1 ; read num2 ;
 #Ελεγχος συνθήκης if, elif και else
if   [ $num1 -eq $num2 ]; then
   echo "Οι δυο αριθμοί είναι ίσοι“
elif [ $num1 -gt $num2 ]; then
   echo "Πρώτος είναι μεγαλύτερος από τον δεύτερο“
else
   echo “Ο δεύτερος είναι μεγαλύτερος από τον πρώτο"
fi
```

# Shell script με την for

- Δημιουργούμε και εκτελούμε το αρχείο **test7**

```
#!/bin/bash
 #Ελεγχος συνθήκης εάν υπάρχουν αρχεία στην
    τρέχουσα λίστα και εκτέλεση της loop
for  f in $( ls); do
    echo $f # εκτυπώνει το όνομα του αρχείου
done
```

# Shell script με την while

- Δημιουργούμε και εκτελούμε το αρχείο **test8**

```
#!/bin/bash
metritis=6 # Δημιουργία μετρητή
#Ελεγχος συνθήκης και εκτέλεση της loop
while [  $metritis -gt 0  ];  do
   #Εκτύπωση τιμής
   echo Η τιμή του μετρητή είναι : $metritis
let metritis=metritis-1
#η μαθηματική εντολή let εκχωρεί τιμή για την μεταβλητή
   done
```