

Διάλεξη 1 - Γραμμή Εντολών, Προγράμματα και Μεταγλώττιση

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Εισαγωγή στον Προγραμματισμό

Θανάσης Αυγερινός

Ανακοινώσεις

- 9 και 16 Οκτωβρίου δεν θα έχουμε διάλεξη και εργαστήρια
 - Για αναπληρώσεις θα υπάρξει αντίστοιχη ανακοίνωση μέσω piazza (διάλεξη) ή eclass (εργαστήριο)

Την Προηγούμενη Φορά

- Τι είναι:
 - Υπολογιστής
 - Πρόγραμμα
 - Γλώσσα Προγραμματισμού
- "Τρέξαμε" ένα "Hello World" C Πρόγραμμα

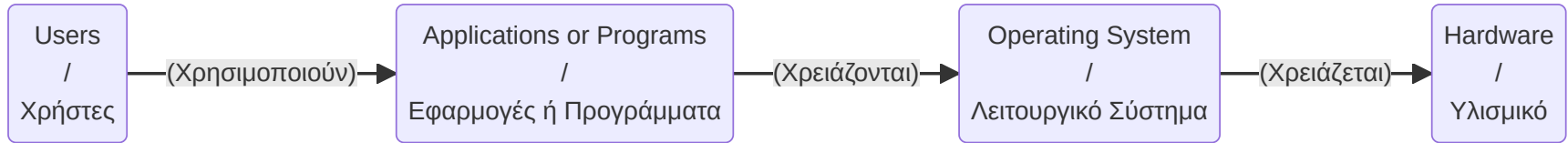
Διάλεξη 1 - Γραμμή Εντολών, Προγράμματα και Μεταγλώττιση

Σήμερα:

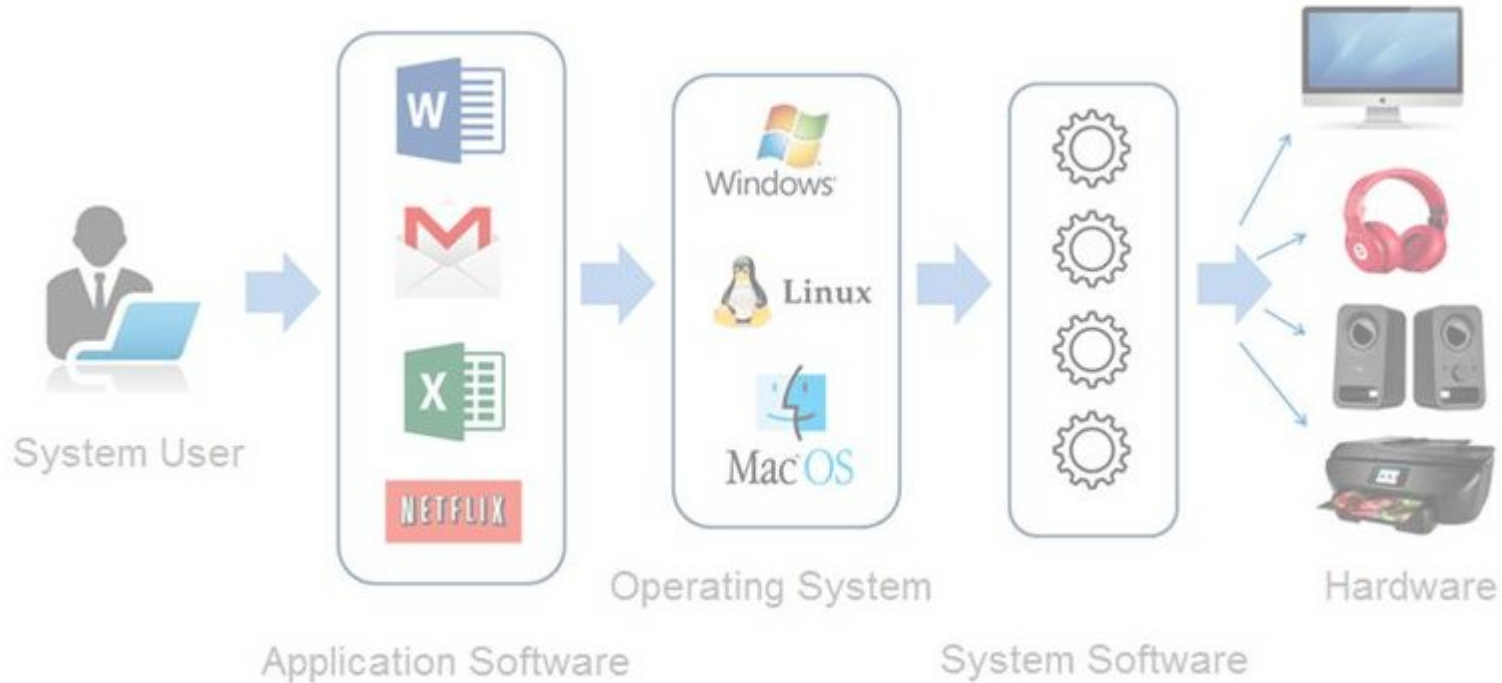
1. Αρχιτεκτονική Υπολογιστικών Συστημάτων
2. Linux και Γραμμή Εντολών
3. Βασικές Εντολές
4. Μεταγλωττιστές
5. Θα μεταγλωττίσουμε και θα τρέξουμε (πραγματικά αυτήν την φορά) Linux προγράμματα

Αρχιτεκτονική Υπολογιστικών Συστημάτων

Βασική Δομή Υπολογιστικών Συστημάτων Σήμερα



Αρχιτεκτονική Υπολογιστικών Συστημάτων



Operating Systems (Λειτουργικά Συστήματα)

Operating System: Το λογισμικό του υπολογιστή που είναι υπεύθυνο για:

- Την διαχείριση του υλικού του υπολογιστή (συσκευές)
- Την διαχείριση πόρων του συστήματος (πόση μνήμη μπορεί να χρησιμοποιηθεί από ένα πρόγραμμα, πότε θα πάρει κύκλους στον επεξεργαστή, κ.ο.κ.)
- Προσφορά υπηρεσιών σε προγράμματα χρηστών (αποθήκευση πληροφοριών στον δίσκο, μεταφορά πληροφοριών μέσω δικτύου, κ.ο.κ.)

Linux

- Open-source operating system βασισμένο στο Unix OS.
- Βασισμένο στο **Linux kernel** (πυρήνα), που ξεκίνησε από τον Linus Torvalds in 1991.
- Γραμμένο κυρίως σε γλώσσα C
- Διαδεδομένο σε χρήση
 - Στηρίζει το 90% των web servers
 - Αποτελεί την βάση του Android - τρέχει στην πλειοψηφία των κινητών
 - Είναι η αποκλειστική επιλογή για supercomputers (υπερυπολογιστές)

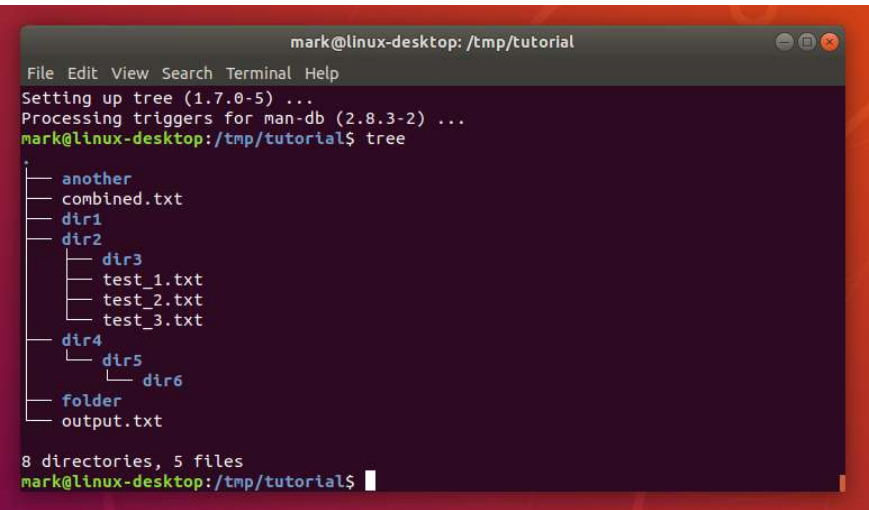
Δύο Τρόποι Διάδρασης με Λειτουργικά Συστήματα

Graphical User Interface (GUI)

Γραφική Διεπαφή Χρήστη

Command Line Interface (CLI)

Διεπαφή Γραμμής Εντολών, Τερματικό (Terminal), Κονσόλα (Console), Κέλυφος (Shell) - Διαφορές





Στο Μάθημα θα επιμείνουμε
στην χρήση CLI

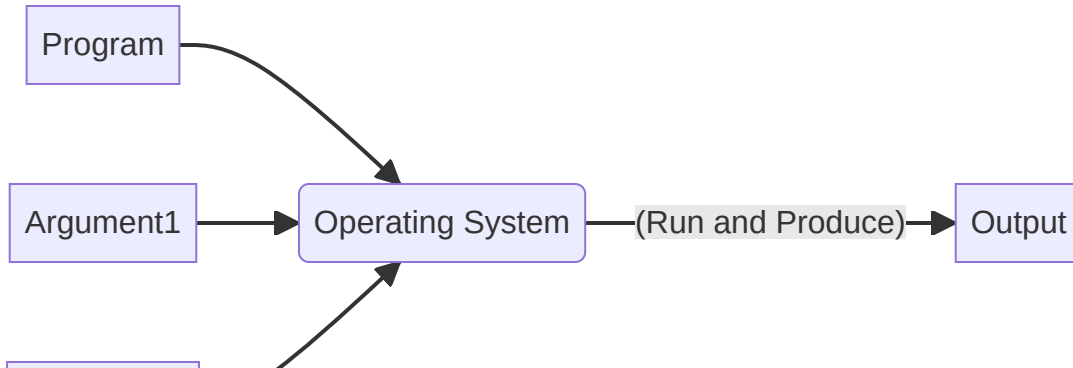
Τρέχοντας Προγράμματα (Εντολές) σε Linux Shell

Για να τρέξουμε ένα πρόγραμμα με N ορίσματα πρέπει να γράψουμε:

Πρόγραμμα **Όρισμα1** **Όρισμα2** ... **ΌρισμαN**

Όρισμα (Argument) / Δεδομένο Εισόδου (Input Data): μια τιμή που μπορεί να χρησιμοποιηθεί από το πρόγραμμα κατά το τρέξιμό του.

Δεδομένο Εξόδου (Output): η τιμή που επιστρέφει το πρόγραμμα όταν τελειώσει τον υπολογισμό του.



Παραδείγματα εντολών

```
thanassis@linux14:~$ who
thanassis pts/0      Oct  5 14:36 (110.220.111.247)
thanassis@linux14:~$ last
...
thanassis@linux14:~$ whoami
thanassis
thanassis@linux14:~$ id
uid=2622(thanassis) gid=1001(dep) groups=1001(dep)
thanassis@linux14:~$ users
thanassis
thanassis@linux14:~$ factor 2394872891
2394872891: 3259 734849
thanassis@linux14:~$ hostname
linux14
thanassis@linux14:~$ sleep 3
thanassis@linux14:~$ uptime
 14:40:23 up 7 days,  5:49,  3 users,  load average: 0.00, 0.00, 0.00
thanassis@linux14:~$ uname -a
Linux linux14 5.4.0-163-generic #180-Ubuntu SMP Tue Sep 5 13:21:23 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux
# Interactive commands follow
thanassis@linux14:~$ top
thanassis@linux14:~$ man
thanassis@linux14:~$ ps
```

Τέσσερις Τρόποι Πρόσβασης *nix Γραμμής Εντολών

Ταξινομημένα από τον πιο πρακτικό προς τον λιγότερο πρακτικό τρόπο για προγραμματισμό

1. Εγκαθιστάς Linux (ή macOS) στον υπολογιστή σου
2. Εγκαθιστάς WSL 2 στο Windows μηχανήμά σου
3. Κάνεις ssh (μέσω putty ή κονσόλας) στα μηχανήματα του εργαστηρίου linuxXY.di.uoa.gr
4. Συνδέσου στο google cloud console που έχεις στον προσωπικό σου λογαριασμό

Στα 1, 2, 4 είσαι διαχειριστής (root) στο σύστημά σου. Στο 3 είσαι χρήστης.

A dramatic landscape photograph of a mountain peak, likely Mount Everest, with a live coding session title overlaid. The mountain is partially shrouded in mist and clouds, with the sun casting a warm, golden glow on its upper sections. The foreground shows dark, forested slopes. The text "Live Coding Session" is centered in a white, serif font.

Live Coding Session

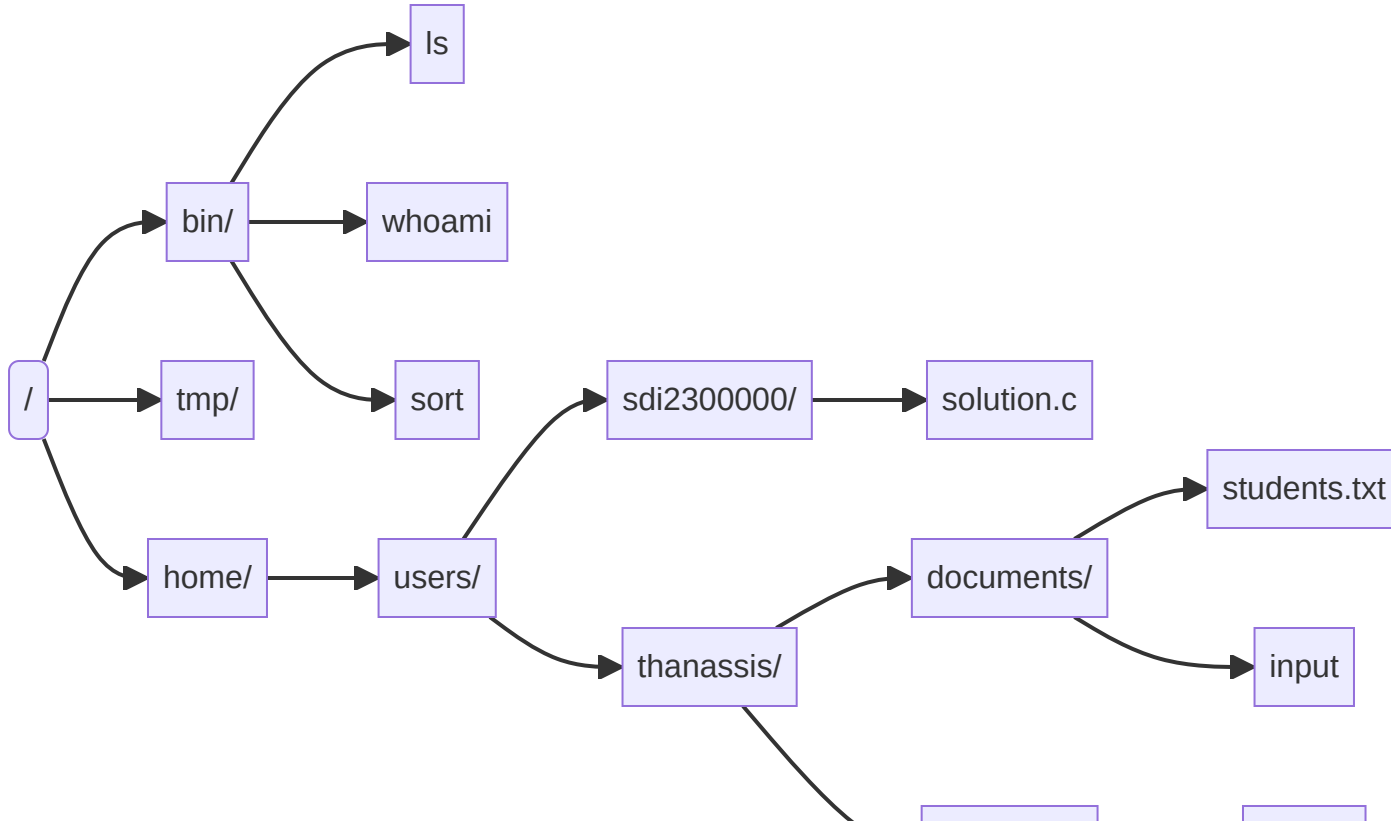
Filesystem (Σύστημα Αρχείων)

Αρχείο (File) είναι ένας πόρος για να καταγράψουμε δεδομένα σε έναν υπολογιστή. Συνήθως αποθηκεύεται στην δευτερεύουσα/μόνιμη μνήμη (πχ σκληρός δίσκος).

- Στο Linux σχεδόν τα πάντα είναι ένα αρχείο.
- Κάθε αρχείο:
 - Έχει ένα όνομα (**filename/basename**)
 - Βρίσκεται μέσα σε ένα συγκεκριμένο φάκελο (**directory/folder**)
 - Έχει ένα πλήρες μονοπάτι (**filepath**) που καθορίζει που βρίσκεται το αρχείο μέσα στο σύστημα αρχείων

Παράδειγμα: το filepath ενός αρχείου είναι ``/home/users/thanassis/documents/students.txt``, ο φάκελος μέσα στον οποίο βρίσκεται αυτό το αρχείο είναι ο ``/home/users/thanassis/documents`` ενώ το όνομα του αρχείου είναι ``students.txt``. Το ``.txt`` στο τέλος του ονόματος λέγεται επέκταση (**extension**) και συνήθως να περιγράφει τον τύπο του αρχείου.

Filesystem Hierarchy (Ιεραρχία Συστήματος Αρχείων)



Βασικές Εντολές Χειρισμού Αρχείων (1/2)

```
# Directory we are currently in
thanassis@linux14:~/examples$ pwd
/home/users/thanassis/examples
# List files
thanassis@linux14:~/examples$ ls
hello.c
# List files with details
thanassis@linux14:~/examples$ ls -l
total 4
-rw-r--r-- 1 thanassis dep 62 Oct  5 16:04 hello.c
# Print contents of file
thanassis@linux14:~/examples$ cat hello.c
#include <stdio.h>

int main() {
    printf("Hello world\n");
}
# Create `foo` directory
thanassis@linux14:~/examples$ mkdir foo
thanassis@linux14:~/examples$ ls
foo hello.c
# New directory is empty by default
thanassis@linux14:~/examples$ ls foo
```

Βασικές Εντολές Χειρισμού Αρχείων (2/2)

```
# Change directory to the newly created one
thanassis@linux14:~/examples$ cd foo
# List files within the directory
thanassis@linux14:~/examples/foo$ ls
# Copy file from parent `..` directory to the current one `.`
thanassis@linux14:~/examples/foo$ cp ../hello.c .
thanassis@linux14:~/examples/foo$ ls
hello.c
# Check the file contents are what we expect
thanassis@linux14:~/examples/foo$ cat hello.c
#include <stdio.h>
```

```
int main() {
    printf("Hello world\n");
}
```

```
# Remove the file
thanassis@linux14:~/examples/foo$ rm hello.c
# Go to the parent directory
thanassis@linux14:~/examples/foo$ cd ..
thanassis@linux14:~/examples$ rmdir foo
thanassis@linux14:~/examples$ ls
hello.c
```

Coreutils (Βασικά Βοηθητικά Προγράμματα)

Τα περισσότερα Linux distros έρχονται με μια σουίτα προγραμμάτων που λέγονται coreutils.

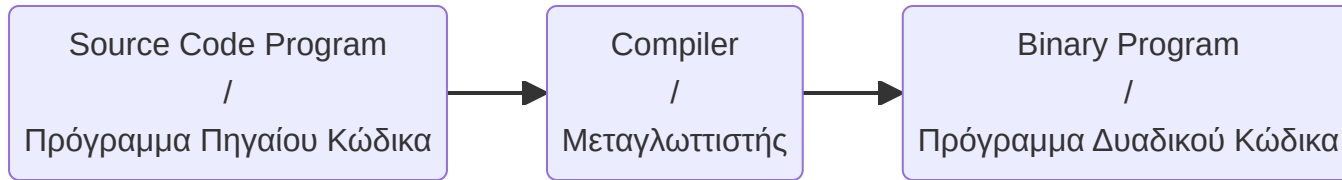
- Αυτά τα προγράμματα είναι ως επί το πλείστον γραμμένα σε C
- Υπάρχουν και άλλα βασικά προγράμματα που δεν κοιτάξαμε (στα coreutils ή εκτός) τα οποία προτείνουμε να εξερευνήσετε
 - π.χ. `find`, `grep`, `sort`, `df -h`, `du -sh`, `wc -l`, `file`, `which`, `basename`, `dirname`, `ping`, `curl` Κ.Ο.Κ.
- Στο live session μιλήσαμε λίγο και για `apt` (πως προσθέτουμε καινούρια προγράμματα) και command line editors (πως αλλάζουμε αρχεία από την κονσόλα). Περισσότερα στα εργαστήρια.



Compilers

Compilers (Μεταγλωττιστές)

Compiler (μεταγλωττιστής) είναι ένα πρόγραμμα που μετατρέπει εντολές μιας γλώσσας προγραμματισμού σε κώδικα μηχανής ώστε να μπορεί να διαβαστεί και να τρέξει από τον υπολογιστή.



Η παραπάνω εικόνα είναι σωστή, αλλά λείπουν κάποιες λεπτομέρειες που θα προσθέσουμε σε επόμενες διαλέξεις. Στο μάθημα θα χρησιμοποιήσουμε τον GNU C Compiler ή αλλιώς `gcc`.

Ας μεταγλωττίσουμε το Hello World!

```
/* File: helloworld.c */
#include <stdio.h>

int main() {
    printf("Hello world\n");
    return 0;
}
```

Ανοίγουμε κονσόλα και τρέχουμε:

```
thanassis@linux14:~/examples$ gcc hello.c
thanassis@linux14:~/examples$ ls
a.out  hello.c
thanassis@linux14:~/examples$ ./a.out
Hello world
thanassis@linux14:~/examples$ file a.out
a.out: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
thanassis@linux14:~/examples$ gcc -o hello hello.c
thanassis@linux14:~/examples$ ./hello
Hello world
```

Ανάλυση του Hello World 1/3

```
/* File: helloworld.c */  
// Another single line comment
```

- **Σχόλια:** κείμενο του προγραμματιστή που συνοδεύει τον κώδικα για να τον κάνει περισσότερο σαφή για τρίτους ή και εμάς τους ίδιους, ειδικά αν έχει περάσει καιρός από τότε που γράψαμε τον κώδικα :)

```
#include <stdio.h>
```

- **#include Directive (Οδηγία):** Με την οδηγία `#include <stdio.h>` ο μεταγλωττιστής συμπεριλαμβάνει (include) τα περιεχόμενα του αρχείου `stdio.h` (standard input output) στον κώδικα του προγράμματος. Το αρχείο `stdio.h` περιέχει τις βασικές (standard) δηλώσεις των συναρτήσεων με τις οποίες γίνεται εμφάνιση δεδομένων στην οθόνη (output) και εισαγωγή δεδομένων από το πληκτρολόγιο (input).

```
printf("Hello world\n");
```

Ανάλυση του Hello World 2/3

- **printf:** Η συνάρτηση βιβλιοθήκης `printf` δηλώνεται μέσα στο αρχείο `stdio.h` (για αυτό κάνουμε `include`) και επιτρέπει την εκτύπωση του αλφαριθμητικού "Hello world\n". Ο χαρακτήρας `'\n'` δημιουργεί μια νέα γραμμή μετά την εμφάνιση του μηνύματος στην οθόνη.

Ανάλυση του Hello World 3/3

```
int main() {  
    return 0;  
}
```

- Ένα πρόγραμμα C ορίζεται από ένα σύνολο **συναρτήσεων**.
- Προκειμένου να μπορούμε να το τρέξουμε, πρέπει να έχει *ακριβώς μία* συνάρτηση `main`, η οποία καλείται πρώτη όταν αρχίσουμε να τρέχουμε το πρόγραμμα.
- Η εντολή `return 0` επιστρέφει την τιμή της συνάρτησης όταν αποτιμηθεί. Η τιμή που επιστρέφει η `main` είναι επίσης και το *exit code* του προγράμματος, δηλαδή η τιμή που δείχνει αν το πρόγραμμα ολοκληρώθηκε με επιτυχία ή όχι. Τρέχοντας `echo $?` σε ένα Linux shell μπορούμε να δούμε την τιμή με την οποία επέστρεψε το πρόγραμμα. Τιμές διάφορες του 0 σημαίνουν ότι το πρόγραμμα απέτυχε.
- Οι εντολές (statements) της συνάρτησης περιέχονται μέσα σε άγκιστρα `{ }` και η κάθε μία τελειώνει με `;` (semicolon / ελληνικό ερωτηματικό).

Για την επόμενη φορά

- Ολοκληρώστε τις εγγραφές στα εργαλεία του μαθήματος.
- Τρέξετε τις εντολές που είδαμε στους δικούς σας λογαριασμούς.

Ξεκινήστε να δουλεύετε την εργασία!

- Προαιρετικό: [The Unix Programming Environment](#).
- Από τις σημειώσεις του κ. Σταματόπουλου, συνιστώ να διαβάσετε σελίδες 58-60, 98-99.

Ευχαριστώ και καλή μέρα εύχομαι!

Keep coding!