

Διάλεξη 1 - Η Γραμμή Εντολών

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Εισαγωγή στον Προγραμματισμό

Θανάσης Αυγερινός

Ανακοινώσεις / Διευκρινίσεις

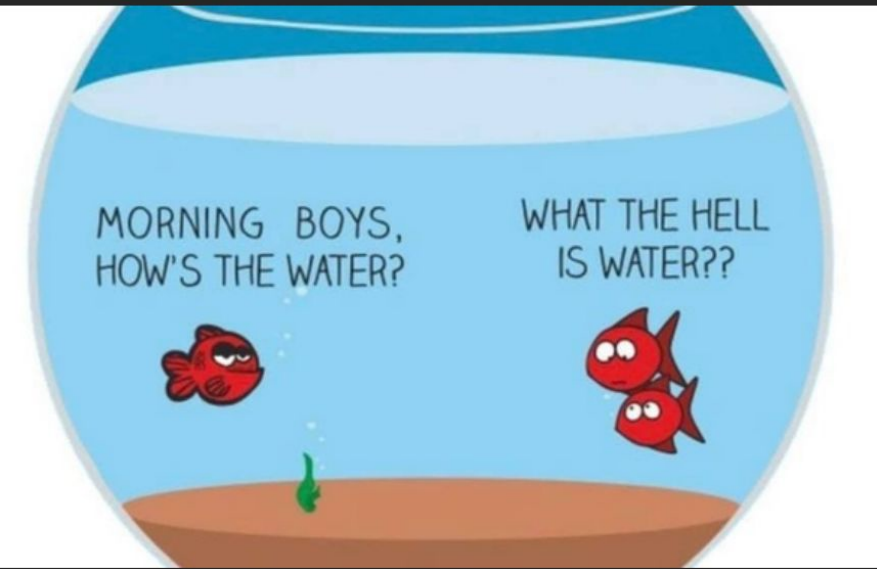
- Η πρώτη εβδομάδα τελείωσε
- Το εργαστήριο ξεκινάει Δευτέρα!



Why do universities still teach C and C++?

There's an old joke. Two fish are swimming along, and another, older fish swims by and says, "Water's nice today, huh?"

A moment later, one of the two fish turns to the other and say, "What's water?"



I'm writing this answer on a Mac. The operating system was written in C.

I'm writing this on Firefox. Most of the core is (AFAIK) written in C++, along with (probably) some Objective-C for the UI layer.

The wireless keyboard and mouse I'm using may have device drivers written in C, and they probably are also themselves programmed in C.

Though some of the specifics might change, the vast majority of software in use today was written in C, C++, or a derivative language. Sure, there's some COBOL, but less each year. And Python is used extensively, but often not for what we would normally call software. And it's not entirely clear how much C heritage is carried in JavaScript, but even so, C and its derivative languages are our water.

And fish should know what water is.

406.8K views · View 2,608 upvotes · View 24 shares · Answer requested by Miguel Paraz

1 of 13 answers

<https://qr.ae/p2wHA3>

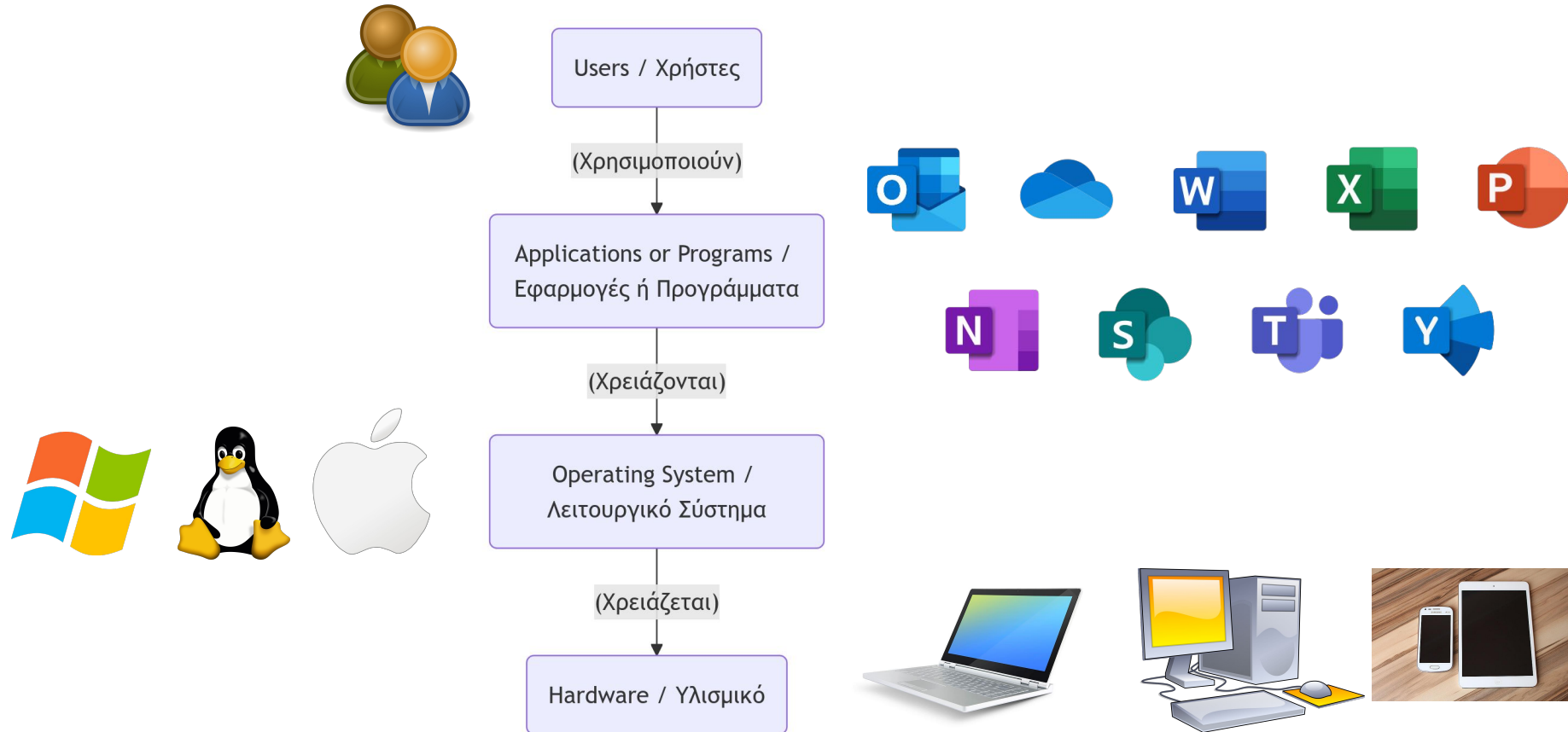
Την Προηγούμενη Φορά

- Καλωσόρισμα
- Computer Science and Programming
- Διαδικαστικά
- Σκοπός του μαθήματος
- Υπολογιστές και προγραμματισμός
- Το πρώτο μας πρόγραμμα - hello world!

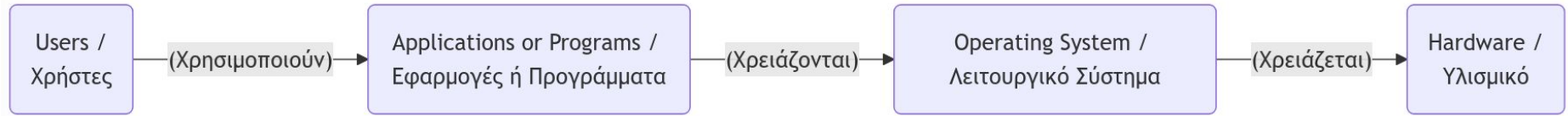
Σήμερα

- Αρχιτεκτονική Υπολογιστικών Συστημάτων
- Linux και Γραμμή Εντολών
- Βασικές Εντολές
- Μεταγλωττιστές
- Θα μεταγλωττίσουμε και θα τρέξουμε (πραγματικά αυτήν την φορά) Linux προγράμματα

Βασική Δομή Υπολογιστικών Συστημάτων Σήμερα

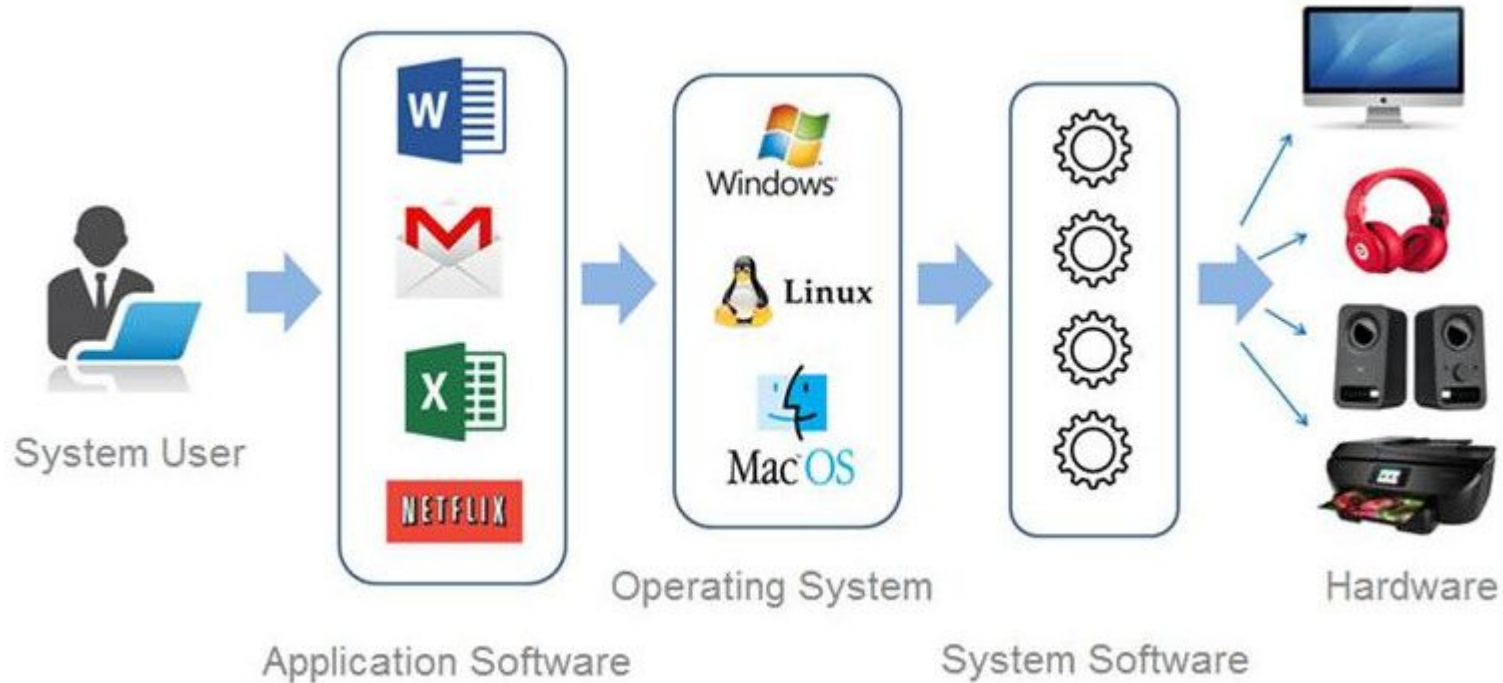


Αρχιτεκτονική Υπολογιστικών Συστημάτων



Παραδείγματα;

Αρχιτεκτονική Υπολογιστικών Συστημάτων



Operating Systems (Λειτουργικά Συστήματα)

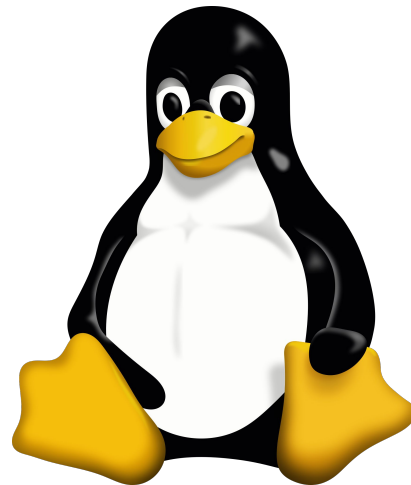
Operating System: Το λογισμικό του υπολογιστή που είναι υπεύθυνο για:

- Την διαχείριση του υλικού του υπολογιστή (συσσκευές)
- 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.)

Γιατί να μην είναι το πρόγραμμά μας υπεύθυνο για αυτά;

Linux

- Open-source operating system βασισμένο στο [Unix](#) OS.
- Βασισμένο στο Linux kernel (πυρήνα), που ξεκίνησε από τον Linus Torvalds το 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)



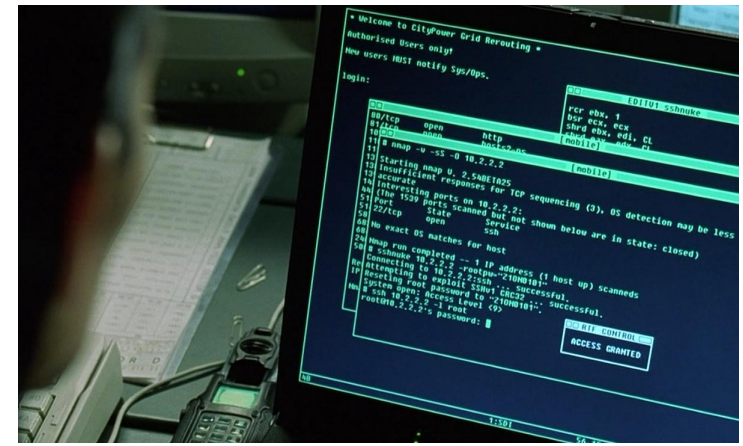
Δύο Τρόποι Διάδρασης με Λειτουργικά Συστήματα

- Graphical User Interface (GUI)
Γραφική Διεπαφή Χρήστη :'(
- Command Line Interface (CLI)
Διεπαφή Γραμμής Εντολών, Τερματικό (Terminal), Κονσόλα (Console), Κέλυφος (Shell) - [Διαφορές](#)



```
mark@linux-desktop: /tmp/tutorial
File Edit View Search Terminal Help
Setting up tree (1.7.0-5) ...
Processing triggers for man-db (2.8.3-2) ...
mark@linux-desktop:/tmp/tutorial$ tree
.
├── another
├── combined.txt
├── dir1
├── dir2
│   ├── dir3
│   │   ├── test_1.txt
│   │   ├── test_2.txt
│   │   └── test_3.txt
├── dir4
│   └── dir5
│       └── dir6
├── folder
└── output.txt

8 directories, 5 files
mark@linux-desktop:/tmp/tutorial$
```



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.

Τρέχοντας Προγράμματα (Εντολές) σε Linux Shell

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

Πρόγραμμα Όρισμα1 Όρισμα2 ... ΌρισμαN

Παράδειγμα:

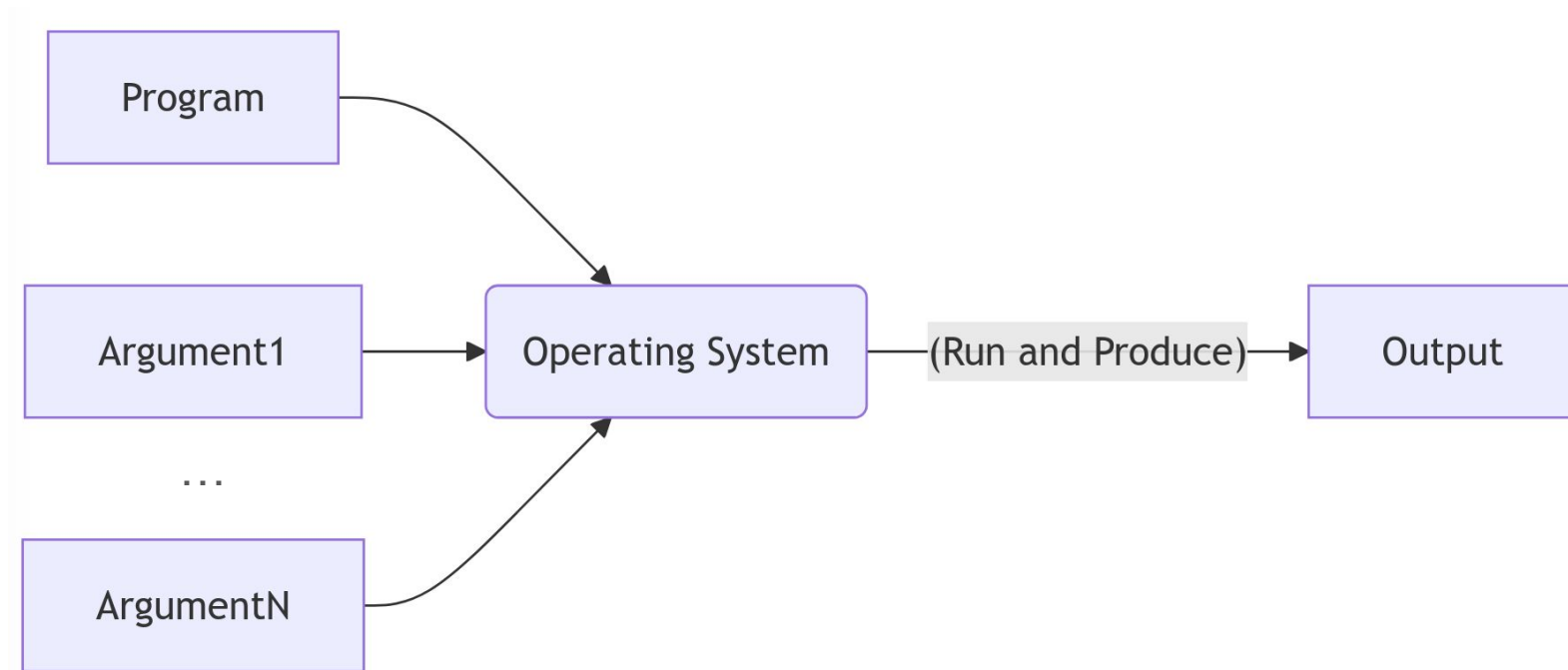
/bin/echo hello good world

Ποιο είναι το πρόγραμμα; Πόσα ορίσματα έχει;

Ας τρέξουμε το πρόγραμμα `/bin/echo`

Τρέχοντας Προγράμματα (Εντολές) σε Linux Shell

Το Shell μας επιτρέπει να τρέξουμε προγράμματα και να πάρουμε outputs:



4 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.

Το "τυφλό" σύστημα / Touch Typing / Blind Typing

Μπορεί να κάνει την ζωή σου πιο εύκολη

- <https://blindtyping.com/>
- <https://www.typingclub.com/>
- <https://www.typingstudy.com/>

Και άλλα πολλά - αναζητήστε πηγές

Stories in the life of a DevOps engineer

Μόλις ήρθε ένα alert



ALARM: "Maximum-CPU-Utilization" in



○ AWS Notifications <no-reply@sns.amazonaws.com>

To: [Redacted]

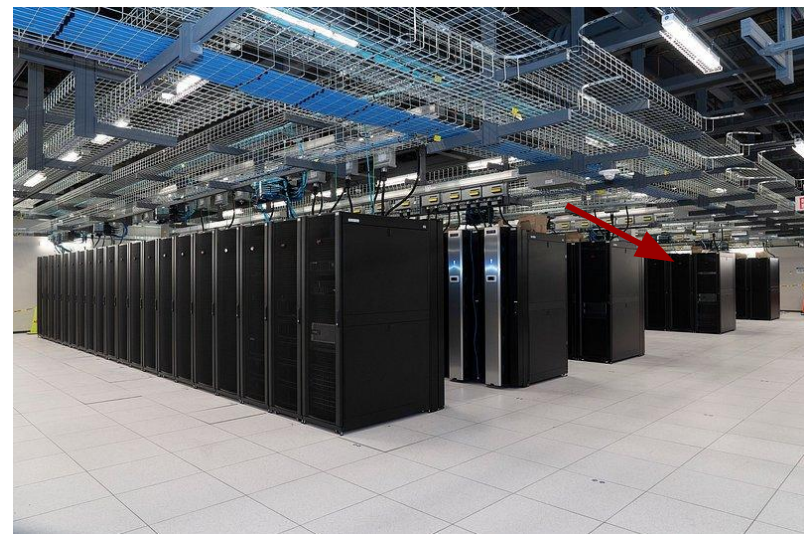
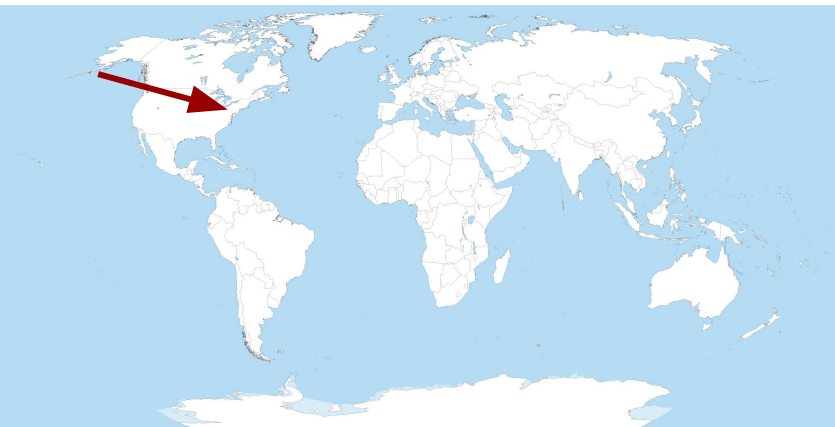
You are receiving this email because your Amazon CloudWatch Alarm "Maximum-CPU-Utilization" is in ALARM state at 20:02:50 UTC.

View this alarm in the AWS Management Console:

[https://us-west-2.console.aws.amazon.com/cloudwatch/home?region=us-west-2:alarm#/alarm-detail/\[Redacted\]](https://us-west-2.console.aws.amazon.com/cloudwatch/home?region=us-west-2:alarm#/alarm-detail/[Redacted])

Alarm Details:

- Name: Maximum-CPU-Utilization
- Description: A CloudWatch Alarm that triggers when Maximum CPU Utilization Exceeds
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 datapoint [8.175136252270871 (23/05/21 19:50:00)] exceeded the threshold (5.0).



Ευτυχώς υπάρχει το "τηλεχειριστήριο": ssh

Το SSH (Secure Shell) είναι ένα πρωτόκολλο δικτύου που επιτρέπει την ασφαλή απομακρυσμένη Σύνδεση και διαχείριση ενός υπολογιστή ή server μέσω κρυπτογραφημένης επικοινωνίας.

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
```

Live Coding Session

Filesystem (Σύστημα Αρχείων)

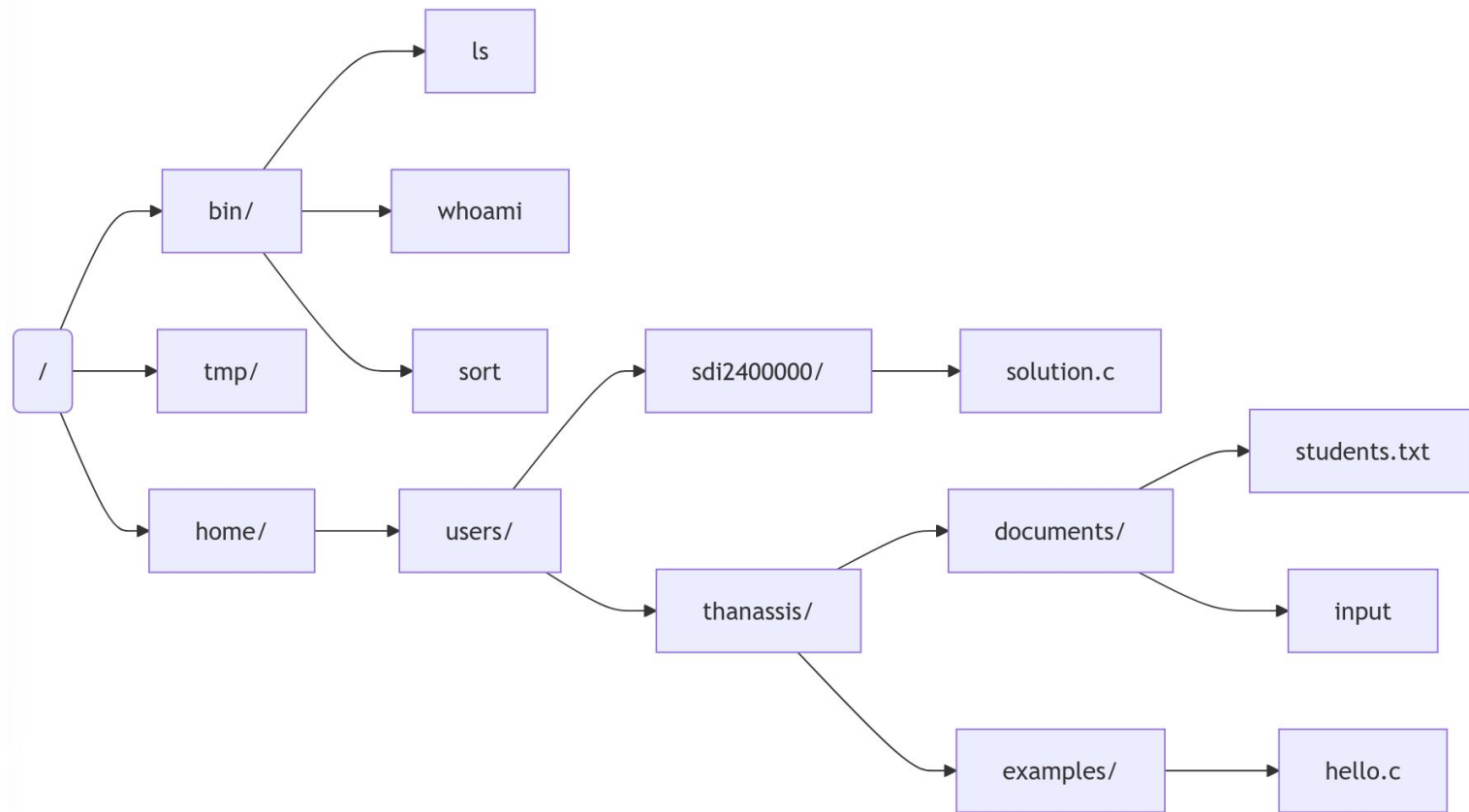
File (Αρχείο) είναι ένας πόρος για να καταγράφουμε δεδομένα σε έναν υπολογιστή. Συνήθως αποθηκεύεται στην δευτερεύουσα/μόνιμη μνήμη (πχ σκληρός δίσκος).

- Στο Linux σχεδόν τα πάντα είναι ένα αρχείο.
- Κάθε αρχείο:
 - Έχει ένα όνομα (**filename/basename**)
 - Βρίσκεται μέσα σε ένα συγκεκριμένο κατάλογο/φάκελο (**directory/folder**)
 - Έχει ένα 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 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 [γγραμμένα 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 (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 προσεγγιστική - λείπουν κάποιες λεπτομέρειες που θα προσθέσουμε σε επόμενες διαλέξεις. Στο μάθημα θα χρησιμοποιήσουμε τον [GNU C Compiler](#) ή αλλιώς gcc.

Ας κάνουμε compile το Hello World!

```
/* File: helloworld.c */
```

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

```
int main() {
```

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

```
    return 0;
```

```
}
```

Ας κάνουμε compile το Hello World!

```
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, BuildID[sha1]=52fa5999c10d767a5ff30f662346478333de74bf, for GNU/Linux 3.2.0, not  
stripped
```

```
thanassis@linux14:~/examples$ gcc -o hello hello.c
```

```
thanassis@linux14:~/examples$ ./hello
```

```
Hello world
```


Ανάλυση του Hello World 1/4

```
/* File: helloworld.c */
```

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

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

Σχόλια (comments): 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 :)

Περιέχεται ανάμεσα στα /* και */

Ή μπορεί να είναι σε μία γραμμή με //:

```
// single line comment
```

Ανάλυση του Hello World 2/4

```
/* File: helloworld.c */
```

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

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

#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).

Ανάλυση του Hello World 3/4

```
/* File: helloworld.c */
```

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

```
int main() {
```

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

```
    return 0;
```

```
}
```

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 4/4 - Όνομα Συνάρτησης

- Ένα πρόγραμμα C ορίζεται από ένα σύνολο **συναρτήσεων** (next time).

```
int main() {
```

```
...
```

```
return 0;
```

```
}
```

Προκειμένου να μπορούμε να το τρέξουμε, πρέπει να έχει **ακριβώς μία** συνάρτηση **main**, η οποία καλείται πρώτη όταν αρχίσουμε να τρέχουμε το πρόγραμμα.

Ανάλυση του Hello World 4/4 - Επιστροφή Συνάρτησης

- Ένα πρόγραμμα C ορίζεται από ένα σύνολο **συναρτήσεων** (next time).

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

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 **ΑΠÉΤΥΧΕ**.

Ανάλυση του Hello World 4/4 - Εντολές Συνάρτησης

- Ένα πρόγραμμα C ορίζεται από ένα σύνολο **συναρτήσεων** (next time).

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

Οι εντολές (statements) της συνάρτησης περιέχονται μέσα σε άγκιστρα `{}` και η κάθε μία τελειώνει με `;` (semicolon / ελληνικό ερωτηματικό).

Kahoot Time!

Για την επόμενη φορά

- Από τις σημειώσεις του κ. Σταματόπουλου σελίδες 58-60, 98-99.
- Ολοκληρώστε τις εγγραφές στα εργαλεία του μαθήματος.
- Γραφτείτε σε κάποιο εργαστήριο!
- Εγκαταστήστε WSL και προσπαθήστε να τρέξετε τις εντολές που είπαμε μόνοι σας!
- Προαιρετικό: [The Unix Programming Environment](#)
- Προαιρετικό #2: [Common Linux Interview Questions](#)

Ευχαριστώ και καλό Σαββατοκύριακο εύχομαι!
Let's Start Coding in Linux ;)