

Προγραμματισμός I

Εισαγωγή

Δημήτρης Μιχαήλ



Τμήμα Πληροφορικής και Τηλεματικής
Χαροκόπειο Πανεπιστήμιο

Βιβλιογραφία

- "C Προγραμματισμός", Deitel & Deitel, Πέμπτη Έκδοση, Εκδόσεις Γκιούρδας, 2010
- "Η Γλώσσα Προγραμματισμού C", B. W. Kernighan, D. M. Ritchie, Δεύτερη Έκδοση, Κλειδάριθμος, 2008

Το μάθημα θα βασιστεί στο πρώτο βιβλίο, το οποίο και προτείνω να διαλέξετε.

Διαφάνειες

Υλικό για το μάθημα θα παρουσιάζεται σταδιακά στις τοποθεσίες:

ηλεκτρονική πλατφόρμα ασύγχρονης τηλεκπαίδευσης

<http://eclass.hua.gr/courses/DIT135>

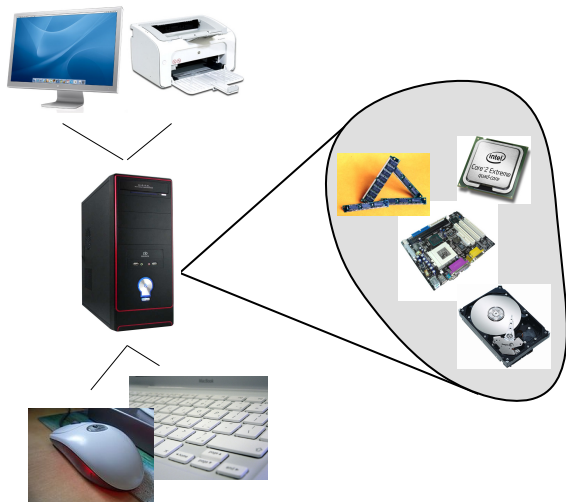
Τι είναι ο υπολογιστής

Ένας υπολογιστής είναι μία συσκευή που μπορεί να εκτελέσει υπολογισμούς και να πάρει λογικές αποφάσεις, με ταχύτητες δισεκατομμύρια φορές μεγαλύτερες από εκείνες των ανθρώπινων όντων.

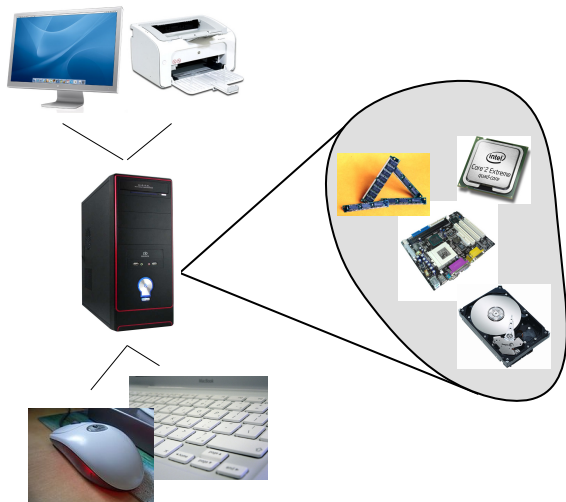
Αλλά για να ξέρει τι να κάνει ένας υπολογιστής, πρέπει κάποιος να του δώσει οδηγίες.

Στο μάθημα αυτό θα μάθουμε να δίνουμε οδηγίες σε έναν υπολογιστή!

Η Ανατομία ενός Υπολογιστή

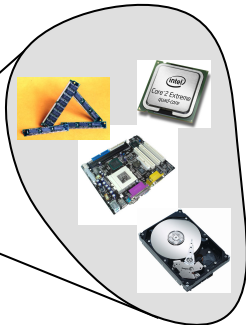


Η Ανατομία ενός Υπολογιστή



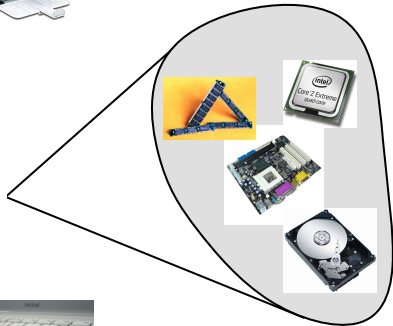
**Συσκευές
Εισόδου:**
πληκτρολόγιο,
ποντίκι, οθόνη
αφής, κ.τ.λ

Η Ανατομία ενός Υπολογιστή



**Συσκευές
Εξόδου:** οθόνη,
εκτυπωτής, κ.τ.λ

Η Ανατομία ενός Υπολογιστή



μονάδα υπολογιστή:
κεντρική μονάδα επεξεργασίας (CPU), μονάδα μνήμης, μονάδα δευτερεύουσας αποθήκευσης (π.χ σκληρός δίσκος), αριθμητική και λογική μονάδα (συνήθως μέσα στη CPU), δίαυλος επικοινωνίας (bus) για επικοινωνία των διαφόρων μονάδων, κ.τ.λ

Λογισμικό

Ο υπολογιστής χρειάζεται και οδηγίες για το τι να κάνει. Αυτό το ρόλο παίζουν τα διάφορα προγράμματα:

- **BIOS:** πρόγραμμα που βοηθάει τον υπολογιστή να ξεκινήσει

Λογισμικό

Ο υπολογιστής χρειάζεται και οδηγίες για το τι να κάνει. Αυτό το ρόλο παίζουν τα διάφορα προγράμματα:

- **BIOS:** πρόγραμμα που βοηθάει τον υπολογιστή να ξεκινήσει
- **λειτουργικό σύστημα:** πρόγραμμα που παίρνει τον έλεγχο μετά την εκκίνηση και διαχειρίζεται όλους τους πόρους του υπολογιστή, τις συσκευές εισόδου και εξόδου και γενικά φροντίζει για την εκτέλεση άλλων προγραμμάτων (π.χ Windows, Linux, SunOS, MacOSX, κ.τ.λ)

Λογισμικό

Ο υπολογιστής χρειάζεται και οδηγίες για το τι να κάνει. Αυτό το ρόλο παίζουν τα διάφορα προγράμματα:

- **BIOS:** πρόγραμμα που βοηθάει τον υπολογιστή να ξεκινήσει
- **λειτουργικό σύστημα:** πρόγραμμα που παίρνει τον έλεγχο μετά την εκκίνηση και διαχειρίζεται όλους τους πόρους του υπολογιστή, τις συσκευές εισόδου και εξόδου και γενικά φροντίζει για την εκτέλεση άλλων προγραμμάτων (π.χ Windows, Linux, SunOS, MacOSX, κ.τ.λ)
- **εφαρμογές γενικής χρήσης:** προγράμματα φτιαγμένα με συγκεκριμένο σκοπό, π.χ αναπαραγωγή ταινιών, επεξεργασία εικόνας, κειμενογράφοι, παιχνίδια, κ.τ.λ

Τι καταλαβαίνει ένας υπολογιστής;

Ένας υπολογιστής καταλαβαίνει από 0 και από 1

Τι καταλαβαίνει ένας υπολογιστής;

Ένας υπολογιστής καταλαβαίνει από 0 και από 1

Αλλά ένας άνθρωπος δε μπορεί εύκολα να μιλάει στον υπολογιστή με 0 και 1

π.χ

000000 00001 00010 00110 00000 100000

σε μία αρχιτεκτονική λέει στον υπολογιστή να προσθέσει τους καταχωρητές 1 και 2 και να βάλει το αποτέλεσμα τον καταχωρητή 6.

Από τις Γλώσσες Μηχανής μέχρι και σήμερα

- γλώσσα μηχανής
- *συμβολομεταφραστικές γλώσσες* (assembly language) η οποία περιέχει συντομεύσεις της αγγλικής γλώσσας, π.χ
ADD32 6, 1, 2

Ένα πρόγραμμα, ο assembler αναλαμβάνει να μετατρέψει την παραπάνω εντολή σε γλώσσα μηχανής.

- *γλώσσες υψηλού επιπέδου* που μας δίνουν την δυνατότητα να γράφουμε οδηγίες που μοιάζουν σχεδόν όπως τα καθομιλουμένα αγγλικά και περιέχουν συνήθη μαθηματικά σύμβολα, π.χ

$$x6 = x1 + x2;$$

Ένα ακόμη πρόγραμμα ο μεταγλωττιστής (compiler) αναλαμβάνει να μετατρέψει το πρόγραμμα υψηλού επιπέδου σε συμβολομεταφραστική γλώσσα.

Η Ιστορία της C

1969-1973	AT & T Bell Labs, Dennis Ritchie
1978	“The C Programming Language” K & R: Kernighan & Ritchie
1983	Σύσταση ANSI Standardization Committee X3J11
1989-1990	Αποδοχή ANSI/ISO Standard (ANSI C)
1990-1999	Αναθεώρηση του standard υπό εξέλιξη C9X (C99)
2011	η έκδοση C11 προσθέτει νέες λειτουργίες (ατομικές λειτουργίες, πολυ-νηματισμός, κ.τ.λ.)
2018	η έκδοση C18 δεν έχει επιπλέον πράγματα πλὴν τεχνικών διορθώσεων της C11

Χαρακτηριστικά της C

- Μικρό μέγεθος

Χαρακτηριστικά της C

- Μικρό μέγεθος
- Χρήση συναρτήσεων

Χαρακτηριστικά της C

- Μικρό μέγεθος
- Χρήση συναρτήσεων
- Χαλαρό σύστημα τύπων (loose typing)

Χαρακτηριστικά της C

- Μικρό μέγεθος
- Χρήση συναρτήσεων
- Χαλαρό σύστημα τύπων (loose typing)
- Δομημένη γλώσσα (structured language)

Χαρακτηριστικά της C

- Μικρό μέγεθος
- Χρήση συναρτήσεων
- Χαλαρό σύστημα τύπων (loose typing)
- Δομημένη γλώσσα (structured language)
- Χαμηλού επιπέδου προγραμματισμός

Χαρακτηριστικά της C

- Μικρό μέγεθος
- Χρήση συναρτήσεων
- Χαλαρό σύστημα τύπων (loose typing)
- Δομημένη γλώσσα (structured language)
- Χαμηλού επιπέδου προγραμματισμός
- Ο προγραμματιστής έχει πλήρη έλεγχο (αλλά και δύναμη) και ευθύνεται για τα σφάλματα του.

Χαρακτηριστικά της C

Η C είναι πάρα πολύ γνωστή για διάφορους λόγους:

- έχει υψηλού επιπέδου δομές
- μπορεί να χρησιμοποιηθεί για χαμηλού επιπέδου προγραμματισμό
- παράγει πολύ αποτελεσματικά προγράμματα
- μπορεί να μεταγλωττιστεί σε πάρα πολλές πλατφόρμες

Οι γλώσσες C++, Java, C#, PHP, κ.τ.λ που θα μάθετε αργότερα όλες βασίζονται στην C.

Δομημένος Προγραμματισμός

Βασίζεται στην οργάνωση του προγράμματος σε μικρές, αυτόνομα κατανοητές, οντότητες. Κάθε τέτοια οντότητα πρέπει να έχει ένα σημείο εισόδου και ένα σημείο εξόδου.

Δομημένος Προγραμματισμός

Βασίζεται στην οργάνωση του προγράμματος σε μικρές, αυτόνομα κατανοητές, οντότητες. Κάθε τέτοια οντότητα πρέπει να έχει ένα σημείο εισόδου και ένα σημείο εξόδου.

Η δομή ενός προγράμματος βασίζεται στις τρεις παρακάτω λογικές δομές:

- 1 ακολουθία (sequence)
- 2 επιλογή (condition)
- 3 επανάληψη (iteration)
- 4 συναρτήσεις (functions)

Με τις παραπάνω δομές μπορεί να εκφραστεί οποιοσδήποτε αλγόριθμος.

Δομημένος Προγραμματισμός

Βασίζεται στην οργάνωση του προγράμματος σε μικρές, αυτόνομα κατανοητές, οντότητες. Κάθε τέτοια οντότητα πρέπει να έχει ένα σημείο εισόδου και ένα σημείο εξόδου.

Η δομή ενός προγράμματος βασίζεται στις τρεις παρακάτω λογικές δομές:

- 1 ακολουθία (sequence)
- 2 επιλογή (condition)
- 3 επανάληψη (iteration)
- 4 συναρτήσεις (functions)

Με τις παραπάνω δομές μπορεί να εκφραστεί οποιοσδήποτε αλγόριθμος.

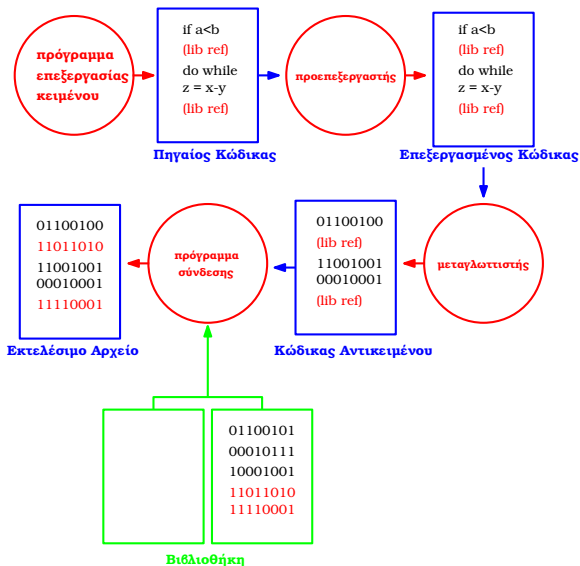
Σε εξαιρετικές περιπτώσεις όπου είναι αναγκαία κάποια παρέκλιση από την χρήση μόνο των παραπάνω, πρέπει να γίνεται τεκμηριωμένα.

Το πρώτο πρόγραμμα σε γλώσσα C

```
1  /*  
2   Hello world program in C  
3   */  
4  #include <stdio.h>  
5  
6  int main()  
7  {  
8      printf( "Hello, world!\n" );  
9  
10     return 0;  
11 }
```

το οποίο λέει στον υπολογιστή να εκτυπώσει στην οθόνη το μήνυμα "hello, world".

Διαδικασία Μεταγλώττισης



Αναλυτική Επεξήγηση Προγράμματος

```
1  /*  
2   Hello world program in C  
3   */
```

Οτιδήποτε είναι ανάμεσα σε

```
/*  
*/
```

είναι σχόλια τα οποία αφαιρούνται τελείως από τον προεπεξεργαστή.

Αναλυτική Επεξήγηση Προγράμματος

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

Δίνει οδηγίες στον μεταγλωττιστή (πιο συγκεκριμένα στον προ-επεξεργαστή) να διαβάσει το αρχείο *stdio.h* το οποίο περιέχει κάποιες βοηθητικές συναρτήσεις για να εκτυπώνουμε στην οθόνη.

Στην προκειμένη περίπτωση την συνάρτηση `printf()` .

Αναλυτική Επεξήγηση Προγράμματος

```
6 int main()  
7 {
```

και

```
11 }
```

είναι ο ορισμός της συνάρτησης `main()` η οποία υπάρχει σε κάθε πρόγραμμα C και λέει στον υπολογιστή από που να ξεκινήσει την εκτέλεση του προγράμματος.

Η συνάρτηση `main()` μόλις τελειώσει την εκτέλεση της επιστρέφει μία τιμή ακεραίου `int` στον χρήστη.

Οι αγκύλες υποδηλώνουν την αρχή και το τέλος της συνάρτησης.

Αναλυτική Επεξήγηση Προγράμματος

8

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

εκτελεί μία συνάρτηση (ορισμένη στο αρχείο που διαβάστηκε με την εντολή `#include <stdio.h>`) η οποία τυπώνει στην οθόνη του υπολογιστή την ακολουθία χαρακτήρων

```
hello, world
```

το `\n` υποδηλώνει έναν ειδικό χαρακτήρα στην C που σημαίνει *newline* (νέα γραμμή) και υποδηλώνει στον υπολογιστή πως πρέπει να αλλάξει γραμμή στην οθόνη.

Ολόκληρη η γραμμή είναι μία **πρόταση** και κάθε πρόταση πρέπει να τερματίζεται πάντα με ένα ελληνικό ερωτηματικό.

Αναλυτική Επεξήγηση Προγράμματος

10

```
return 0;
```

εκτελεί μία πρόταση ακόμη που λέει στο πρόγραμμα να σταματήσει την εκτέλεση την συνάρτησης `main()` και να επιστρέψει στον χρήστη την τιμή 0.

Γενικές Γραμμές Προγραμματισμού

- να χρησιμοποιείτε σχόλια ώστε ο κώδικας να είναι ευανάγνωστος και από 3ους
- μεταξύ των αγκίστρων { } ο κώδικας που εμφανίζεται πρέπει να είναι στοιχισμένος πιο δεξιά (π.χ με 1 tab ή με 8 κενά) ώστε να είναι πιο ευδιάκριτος
- τα προγράμματα σας πρέπει να είναι απλά, δηλαδή να αποφεύγεται η χρήση "προγραμματιστικών κόλπων"

Παράδειγμα προς Αποφυγή

```
1 #include <stdio.h>
2
3 main(t,_,a)
4 char *a;
5 {return!0<t?t<3?main(-79,-13,a+main(-87,1-_,
6 main(-86, 0, a+1 )+a)):1,t<_?main(t+1, _, a ):3,main ( -94, -27+t, a
7 )&&t == 2 ?_<13 ?main ( 2, _+1, "%s %d %d\n" ):9:16:t<0?t<-72?main(.,
8 t, "@n'+, #'/*{w+/w#cdnr/+, {r/*de}+, /*{*, /w{%+, /w#q#n+, /#{l, +, /n{n+\
9 , /+##n+, /#;#q#n+, /+k#; *+, /'r : 'd*'3, }{w+K w'K:'+)e#';dq#'l q#' +d'K#!/\
10 +k#;q#'r}eKK#}w'r}eKK{nl}'/ #;#q#n'}{#}w')}{nl}' /+#n';d}rw' i;# ) {n\
11 l]!/n{n#'; r{#w'r nc{nl}]/#{l,+ 'K {rw' iK{;[{nl}]/w#q#\
12 n'wk nw' iw{k{KK{nl}!/w{%l##w# ' i; : {nl}'/*{q#'ld;r'}{nlwb!/*de}'c \
13 ; ; {nl}'-}{rw}' /+, }##'*)#nc, ', #nw}' /+kd'+e}+; \
14 #'rdq#w! nr' / ' ) }+}{rl#'{n' ' )# }'+}##(!/!/"
15 :t<-50?_==*a ?putchar(a[31]):main(-65,_,a+1):main((*a == '/')+t,_,a\
16 +1 ):0<t?main ( 2, 2 , "%s" ): *a=='/'||main(0,main(-61,*a, "!ek;dc \
17 i@bK' (q)-[w]*#n+r3#l, { }:\nuwloca-O;m .vpbks, fxntdCeghiry"), a+1);}
```

Αυτός ο κώδικας μπορεί να μεταγλωττιστεί με επιτυχία!!