

Διάλεξη 0 - Καλημέρα Κόσμε

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Εισαγωγή στον Προγραμματισμό

Θανάσης Αυγερινός / Τάκης Σταματόπουλος

#whoami

Σήμερα

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



Καλωσορίσατε στο Τμήμα!

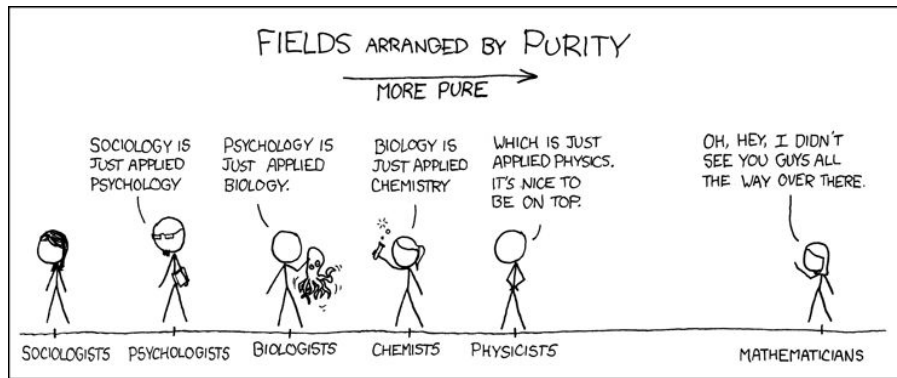
- Στα καλύτερα 250 Πανεπιστήμια του [κόσμου](#) σε Computer Science
- Το πιο βελτιωμένο στην Ευρώπη και [6η θέση παγκοσμίως](#) το 2024-25 σε ποιοτική εκπαίδευση
- Ευκαιρία να συναναστραφείτε με ιδιαίτερα ταλαντούχα άτομα



Γιατί να ασχοληθείς με τον προγραμματισμό το 2025;

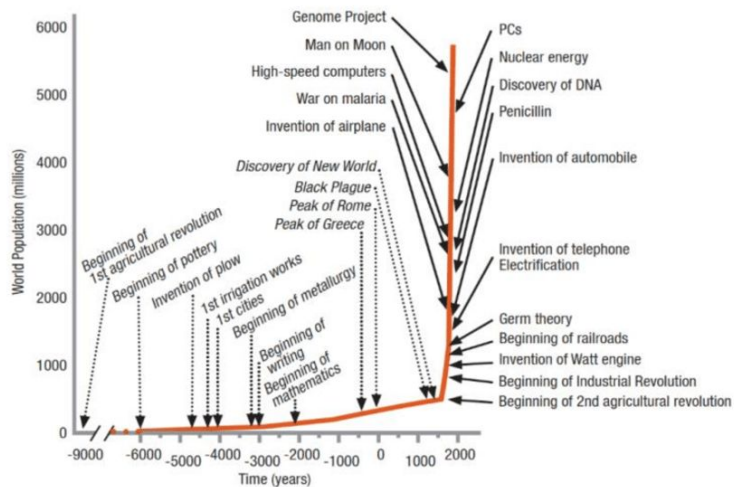
Ενδεικτικοί Λόγοι

- Ευελιξία (θεματική και γεωγραφική)



- Πνευματικά απαιτητική δουλειά

- Στην αιχμή των εξελίξεων



- Cool

Διαδικαστικά

Διαδικαστικά 1/5 - Site Μαθήματος

<https://progintro.github.io>



Δύο τμήματα (άρτιοι / περιττοί) - η ύλη, το εργαστήριο,
ασκήσεις, διαγώνισμα του μαθήματος θα είναι το ίδιο και για τα δύο τμήματα

Διαδικαστικά 2/5 - Ώρες

- Διαλέξεις
 - Δευτέρα και Παρασκευή 9πμ-11πμ
στο Αμφιθέατρο (άρτιοι) και Α2 (περιττοί)
- Ώρες Γραφείου
 - Δευτέρα 11πμ-12μμ
 - Α40 (Αυγερινός)
 - Α48 (Σταματόπουλος)
 - Παρασκευή 11πμ-12μμ
 - Α3 (Αυγερινός)
 - Α48 (Σταματόπουλος)



Διαδικαστικά 3/5 - Βαθμολογία

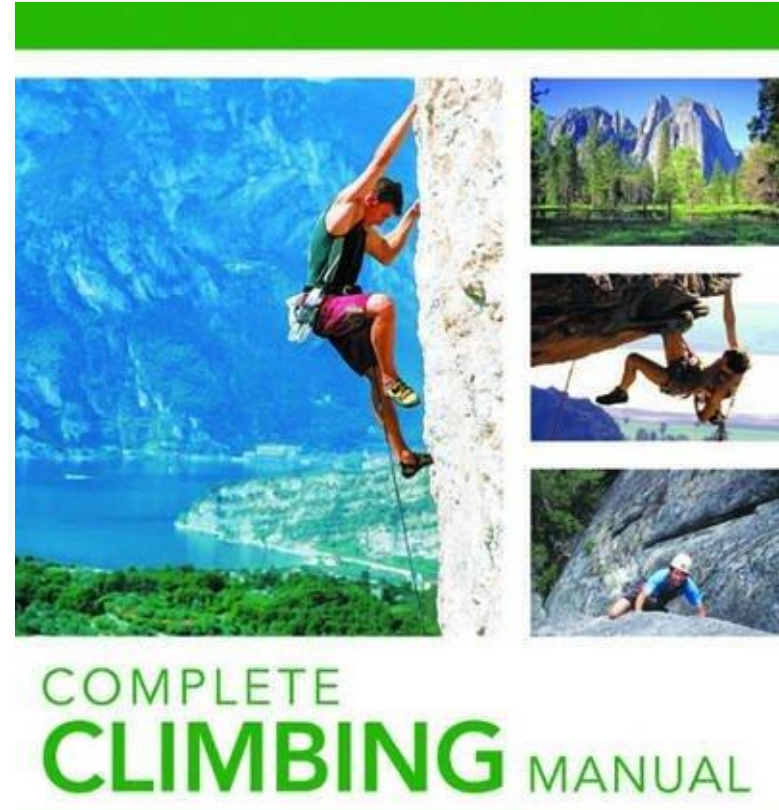
- Αν είστε πρωτοετείς:
 - $50\% * \text{Τελική Εξέταση} + 30\% * \text{Ασκήσεις} + 20\% * \text{Εργαστήριο}$
 - Αλλιώς:
 - $70\% * \text{Τελική Εξέταση} + 30\% * \text{Ασκήσεις}$
 - Αναπροσαρμογή: Αν ο βαθμός από τις **Ασκήσεις** είναι πάνω από 3 μονάδες μεγαλύτερος του βαθμού από την **Τελική Εξέταση**, τότε ο βαθμός για τις **Ασκήσεις** αναπροσαρμόζεται σε **Τελική Εξέταση + 3**.
1. **Τελική Εξέταση**: γραπτή με κλειστά βιβλία
 2. **Ασκήσεις**: προαιρετικές αλλά η επίλυσή τους βοηθάει σημαντικά
 3. **Εργαστήριο**: υποχρεωτική παρουσία για τους πρωτοετείς (μέχρι 2 απουσίες)

Άρα μπορώ να περάσω χωρίς να κάνω εργασίες/εργαστήριο;

In theory, theory and practice are the same.

In practice, they are not.

Θεωρητικά ναι, στην πράξη ... όχι



Διαδικαστικά 4/5 - Εργαστήριο

Διεξάγεται σε **εβδομαδιαία** βάση, ξεκινάμε:

την εβδομάδα της **6ης Οκτωβρίου**

Μπορείτε να γραφτείτε στην Εργαστηριακή Ομάδα που σας ταιριάζει εδώ:

<https://eclass.uoa.gr/modules/group/index.php?course=DI681&urlview=1>

Δεν βρίσκετε κάποιο τμήμα; Μην αγχώνεστε, θα ανοίξουν και άλλες θέσεις!

Διαδικαστικά 5/5 - Εργαλεία του Μαθήματος

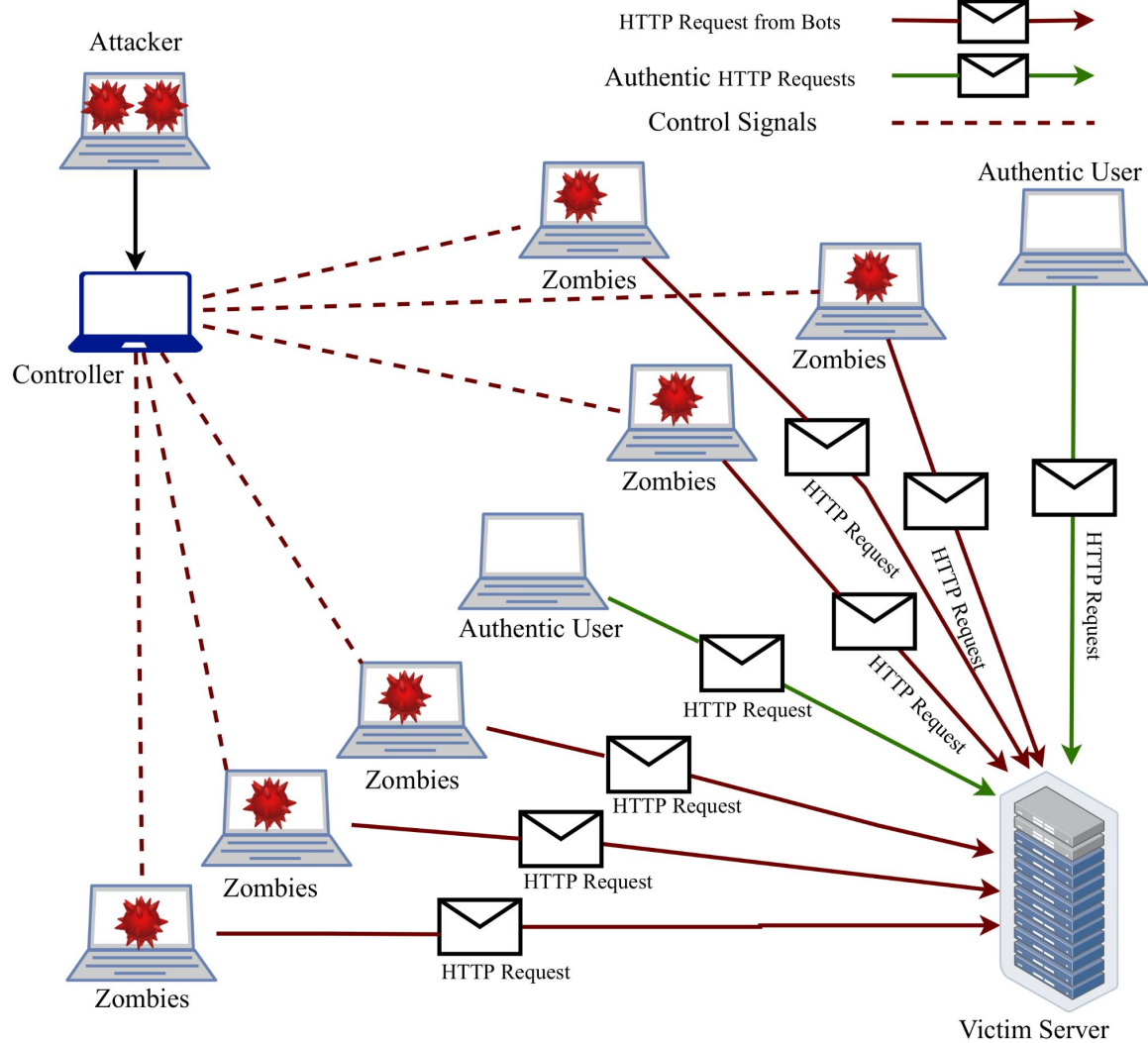
1. Τον προσωπικό σας λογαριασμό email στο [Gmail](#)
2. Τον προσωπικό σας λογαριασμό στο [GitHub](#)
 - a. Επιλέξτε ένα cool αλλά και επαγγελματικό ψευδώνυμο
3. Τον ακαδημαϊκό σας λογαριασμό μέσω <http://webadm.uoa.gr/>
4. Να γραφτείτε στο [Piazza](#) για την επικοινωνία του μαθήματος
5. Να γραφτείτε σε τμήμα εργαστηρίου στο [eclass](#) για τα εργαστήρια
6. Συμπληρώστε την [φύρμα](#) του μαθήματος με τα στοιχεία σας

Συνιστώ να έχετε πρόσβαση σε υπολογιστή κατά την διάρκεια του μαθήματος προκειμένου να προγραμματίζετε ζωντανά.

Browsing Time

progintro, sign ups, φόρμα, σημειώσεις, συγγράμματα, delos

Denial of Service Attacks

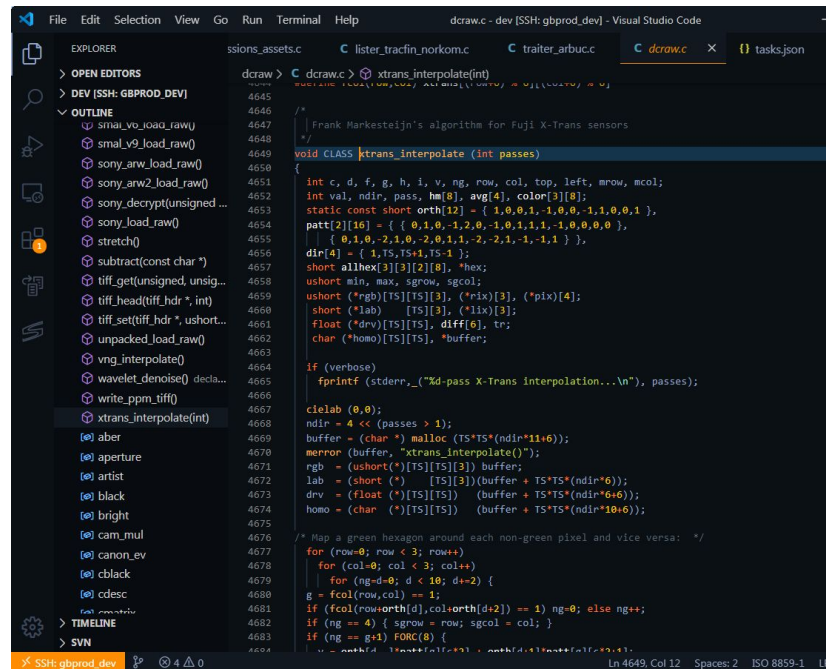


Έχω μια ερώτηση - τι κάνω;

- A. Στέλνω email στον Αυγερινό/Τάκη
- B. Ελέγχω αν έχει απαντηθεί στο piazza και εφόσον *δεν* έχει απαντηθεί, ποστάρω καινούρια ερώτηση

Σκοπός του Μαθήματος

- Εισαγωγή στις βασικές αρχές του προγραμματισμού και την αλγοριθμική σκέψη
- Εμπειρία σε γραφή κώδικα και αποσφαλμάτωση (debugging) σε γλώσσα προγραμματισμού C
- Εξοικείωση με τα "εργαλεία της δουλειάς"



```
File Edit Selection View Go Run Terminal Help
ddraw.c - dev [SSH: gbprod_dev] - Visual Studio Code

EXPLORER
> OPEN EDITORS
> DEV [SSH: GBPROD_DEV]
OUTLINE
  smal_vo_load_raw()
  smal_v9_load_raw()
  sony_arw_load_raw()
  sony_arw2_load_raw()
  sony_decrypt(unsigned ...)
  sony_load_raw()
  stretch()
  subtract(const char *)
  tiff_get(unsigned, unsig...
  tiff_head(tiff_hdr *, int)
  tiff_set(tiff_hdr *, ushort...
  unpacked_load_raw()
  vng_interpolate()
  wavelet_denoise() decla...
  write_ppm_tiff()
  xtrans_interpolate(int)
  aber
  aperture
  artist
  black
  bright
  cam_mul
  canon_ev
  cblack
  cdesc
  ...
> TIMELINE
> SVN

ssions_assets.c
ddraw.c
lister_tracfin_norkom.c
ctrailer_arbucc
ddraw.c
tasks.json

/*
 * Frank Marksteijn's algorithm for Fujif X-Trans sensors
 */
void CLASS xtrans_interpolate (int passes)
{
    int c, d, f, g, h, i, v, ng, row, col, top, left, mrow, mcol;
    int val, ndir, pass, hm[8], avg[4], color[3][8];
    static const short orth[12] = { 1,0,0,1,-1,0,0,-1,1,0,0,1 },
    patt[2][16] = { { 0,1,0,-1,2,0,-1,0,1,1,1,-1,0,0,0,0 },
    { 0,1,0,-2,1,0,-2,0,1,1,-2,-2,1,-1,-1,1 } },
    dir[4] = { 1,15,15,1;
    short allhex[3][3][2][8], *hex;
    ushort min, max, sgrow, sgcol;
    ushort (*rgb)[TS][TS][3], (*rix)[3], (*pix)[4];
    short (*lab) [TS][3], (*lix)[3];
    float (*drv)[TS][TS], diff[8], tr;
    char (*homo)[TS][TS], *buffer;

    if (verbose)
        fprintf (stderr, ("Xd-pass X-Trans Interpolation...\n"), passes);

    cielab (0,0);
    ndir = 4 << (passes > 1);
    buffer = (char *) malloc (TS*TS*(ndir*11+6));
    merror (buffer, "xtrans_interpolate()");
    rgb = (ushort (*)(TS)[TS][3]) buffer;
    lab = (short (*)(TS)[3])(buffer + TS*TS*(ndir*6));
    drv = (float (*)(TS)[TS]) (buffer + TS*TS*(ndir*6+6));
    homo = (char (*)(TS)[TS]) (buffer + TS*TS*(ndir*10+6));

    /* Map a green hexagon around each non-green pixel and vice versa: */
    for (row=0; row < 3; row++)
        for (col=0; col < 3; col++)
            for (ng=d=0; d < 10; d+=2) {
                g = fcol(row,col) == 1;
                if (fcol(row+orth[d],col+orth[d+2]) == 1) ng=0; else ng++;
                if (ng == 4) { sgrow = row; sgcol = col; }
                if (ng == 4+1) FORC(8) {
                    ...
                }
            }
    ...
}
```

Με άλλα λόγια: junior software developer με προοπτικές

Περιεχόμενο Μαθήματος

	Διάλεξη 0: Καλημέρα Κόσμε!
	Διάλεξη 1: Η Γραμμή Εντολών
lab00	Διάλεξη 2: Μνήμη και Μεταβλητές
	Διάλεξη 3: Συναρτήσεις
lab01	Διάλεξη 4: Git και Τελεστές
	Διάλεξη 5: Τελεστές και Εντολές
lab02	Διάλεξη 6: Εντολές και Ροή Ελέγχου
	Διάλεξη 7: Επίλυση Προβλημάτων
lab03	Διάλεξη 8: Ροή Ελέγχου #2
	Διάλεξη 9: Δεδομένα Εισόδου
lab04	Διάλεξη 10: Πίνακες
	Διάλεξη 11: Δείκτες και Αναδρομή
lab05	Διάλεξη 12: Δείκτες και Πίνακες
	Διάλεξη 13: Μνήμη
lab06	Διάλεξη 14: Εμβέλεια, Μνήμη και Συμβολοσειρές
	Διάλεξη 15: Επίλυση Προβλημάτων
lab07	Διάλεξη 16: Πολυπλοκότητα και Προεπεξεργαστής
	Διάλεξη 17: Δυναμική Αναζήτηση
lab08	Guest Lecture on Memory Managment
	Διάλεξη 18: Ταξινόμηση και Δεδομένα Εισόδου #2
lab09	Διάλεξη 19: Δομές
	Διάλεξη 20: Προχωρημένες Δομές
lab10	Διάλεξη 21: Λίστες και Δέντρα
	Guest Lecture on Integer Factorization Algorithms
lab11	Διάλεξη 22: Οργάνωση Κώδικα
	Διάλεξη 23: Προχωρημένα Θέματα

Βασικά / Διαδικαστικά

hw0

Πρώτα Προγράμματα

hw1

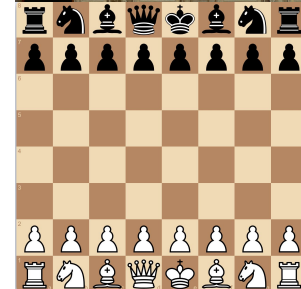
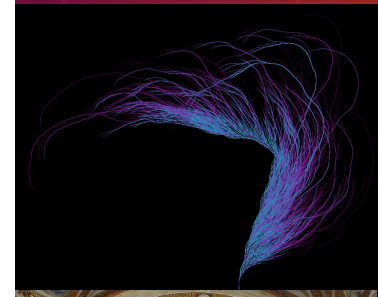
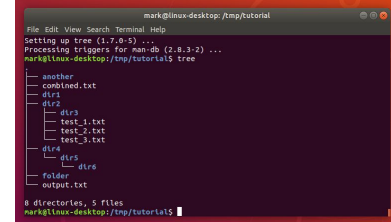
Σύνθετα Προγράμματα

hw2

Αλγόριθμοι και Δομές

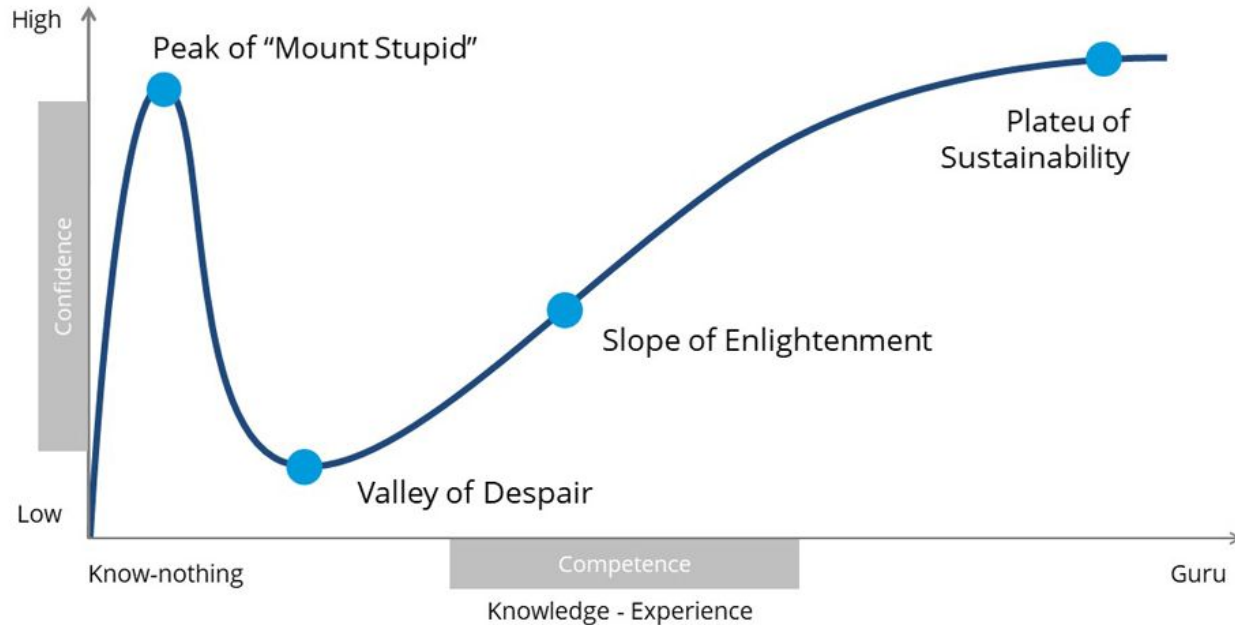
Προχωρημένα Θέματα

project



Ακούγονται πολλά; Ένα βήμα την φορά!

Dunning-Kruger Effect Curve



Τι είναι ο Υπολογιστής;

Μια κατασκευή που έχει την ικανότητα να επεξεργάζεται ένα σύνολο από δεδομένα που του δίνονται και να παράγει τα απαιτούμενα αποτελέσματα.

Γιατί χρησιμοποιούμε υπολογιστές;

1. Ταχύτητα στην επεξεργασία δεδομένων
2. Μνήμη διαθέσιμη για αποθήκευση δεδομένων

Τι είναι ο Προγραμματισμός;

Προγραμματισμός είναι ο σαφής καθορισμός μίας διαδικασίας, σαν ένα σύνολο από εντολές (το πρόγραμμα), που περιγράφει λεπτομερώς τα βήματα που πρέπει να γίνουν για να επιλυθεί ένα πρόβλημα υπολογισμού.

Roll: Έχει κάποιο παιδί προγραμματίσει;

Roll: Έχει κάποιο παιδί προγραμματίσει;

Κάποια παραδείγματα

προγραμμάτων που ξέρετε;

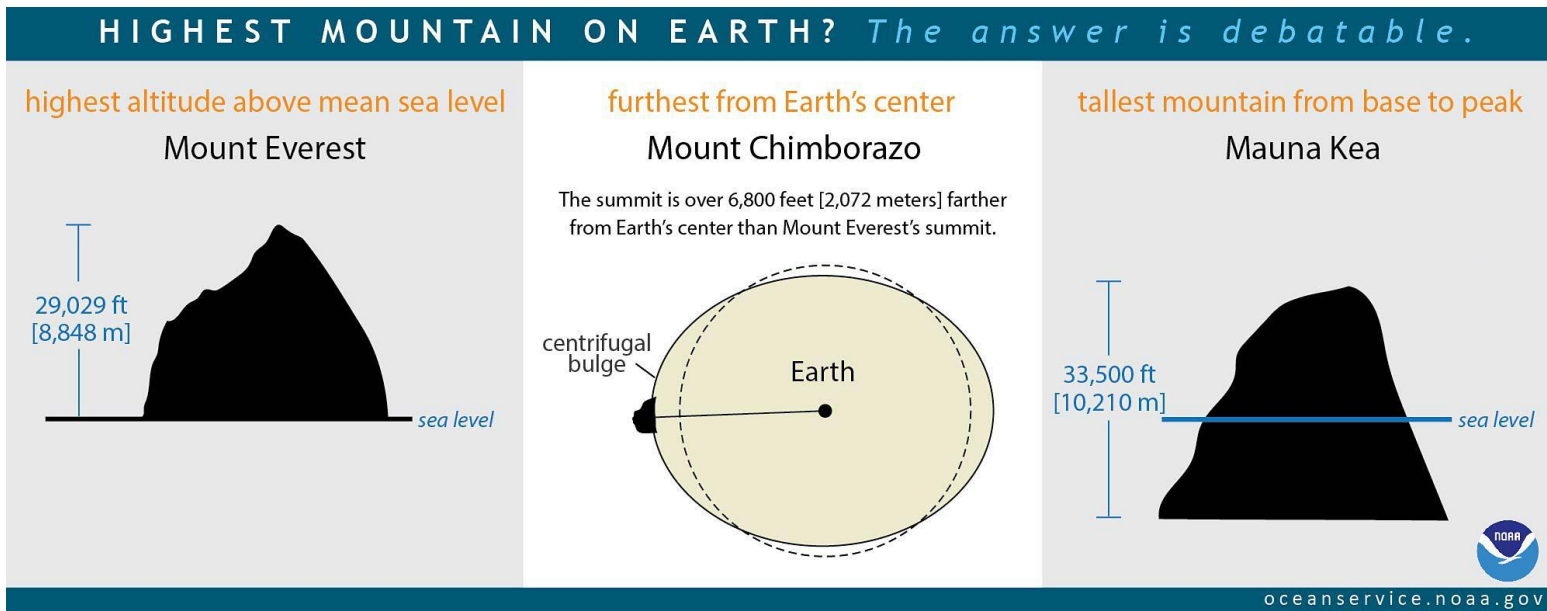
Roll: Έχει κάποιο παιδί προγραμματίσει;

Κάποια παραδείγματα
προγραμμάτων που ξέρετε;

Πρόγραμμα είναι η καταγραφή της
επίλυσης ενός προβλήματος



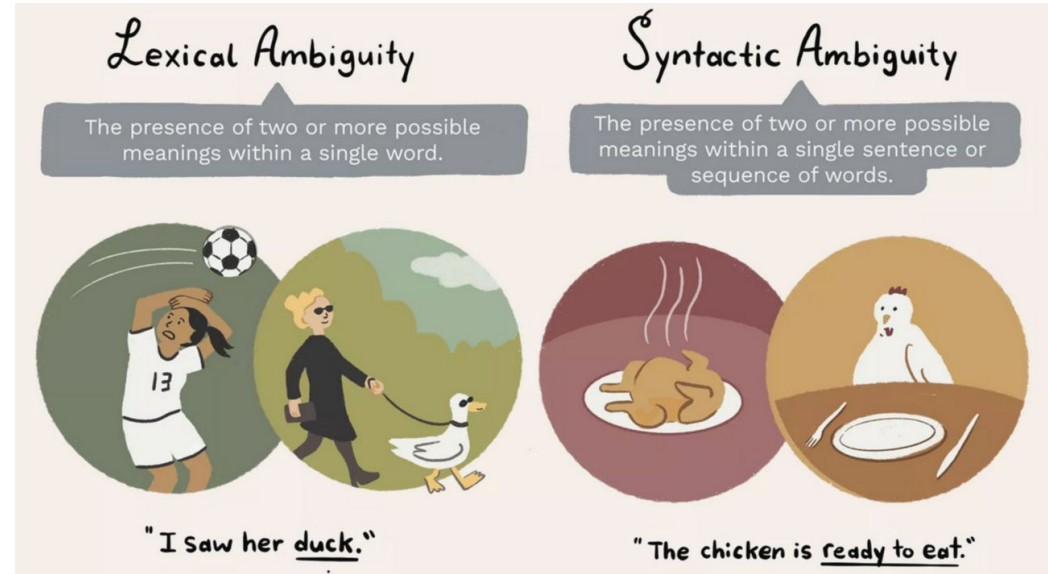
Quiz: Ποιο είναι το πιο ψηλό βουνό του κόσμου;



Η ανθρώπινη γλώσσα έχει αμφισημίες

Γιατί επινοήσαμε τις Γλώσσες Προγραμματισμού;

- > Joe says to his programmer friend Charlie:
- > "Go to the store and buy a loaf of bread. If they have eggs, buy a dozen."
- > Charlie returns with 12 loaves of bread.



Τι είναι η Γλώσσα Προγραμματισμού;

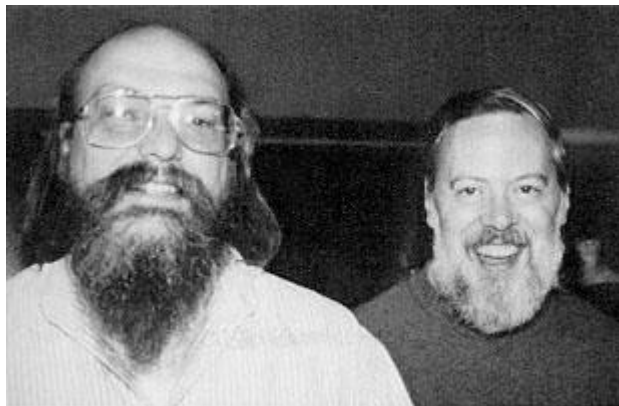
Μια γλώσσα που μας επιτρέπει να επικοινωνούμε εντολές στον υπολογιστή. Μια καλή γλώσσα προγραμματισμού δεν επιτρέπει αμφισημία.

Bonus: Η γλώσσα προγραμματισμού σε συνδυασμό με τον υπολογιστή μπορεί να χρησιμοποιηθεί για να επιλύσει αμφισημίες ανάμεσα σε προγραμματιστές.

Γιατί η Γλώσσα Προγραμματισμού C;

- Απαιραίτητη στο πρόγραμμα σπουδών
- Δημοφιλής
- Χρησιμοποιείται παντού και (σχεδόν) για τα πάντα

Ken Thompson
(co-creator
UNIX and B)



Dennis Ritchie
(creator of C,
co-creator of
UNIX and B)

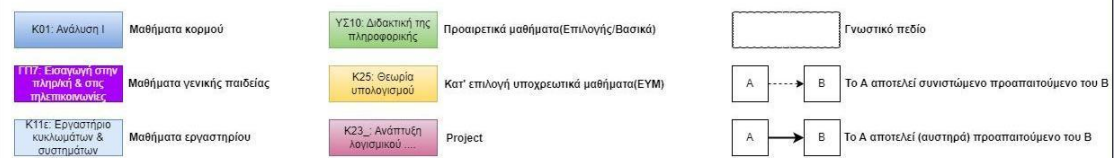


~1970

you are here



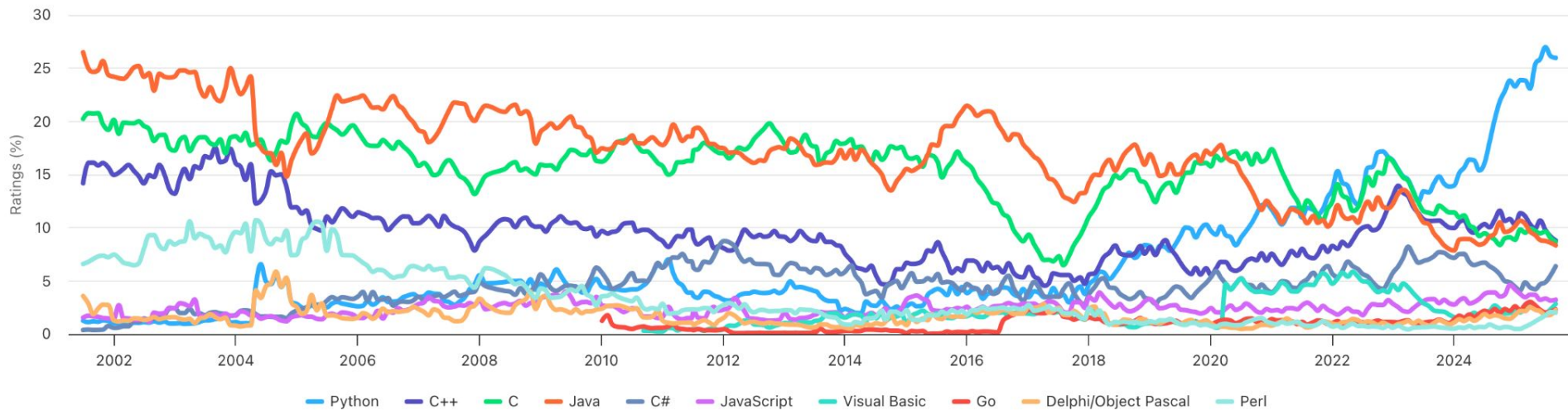
ΥΠΟΜΝΗΜΑ



Στις πιο δημοφιλείς Γλώσσες από το 1970 [[tiobe.com](https://www.tiobe.com)]

TIOBE Programming Community Index

Source: www.tiobe.com



Γλώσσα C: Το "αγωνιστικό" των Γλωσσών

	Energy
(c) C	1.00
(c) Rust	1.03
(c) C++	1.34
(c) Ada	1.70
(v) Java	1.98
(c) Pascal	2.14
(c) Chapel	2.18
(v) Lisp	2.27
(c) Ocaml	2.40
(c) Fortran	2.52
(c) Swift	2.79
(c) Haskell	3.10
(v) C#	3.14
(c) Go	3.23
(i) Dart	3.83
(v) F#	4.13
(i) JavaScript	4.45
(v) Racket	7.91
(i) TypeScript	21.50
(i) Hack	24.02
(i) PHP	29.30
(v) Erlang	42.23
(i) Lua	45.98
(i) Jruby	46.54
(i) Ruby	69.91
(i) Python	75.88
(i) Perl	79.58

	Time
(c) C	1.00
(c) Rust	1.04
(c) C++	1.56
(c) Ada	1.85
(v) Java	1.89
(c) Chapel	2.14
(c) Go	2.83
(c) Pascal	3.02
(c) Ocaml	3.09
(v) C#	3.14
(v) Lisp	3.40
(c) Haskell	3.55
(c) Swift	4.20
(c) Fortran	4.20
(v) F#	6.30
(i) JavaScript	6.52
(i) Dart	6.67
(v) Racket	11.27
(i) Hack	26.99
(i) PHP	27.64
(v) Erlang	36.71
(i) Jruby	43.44
(i) TypeScript	46.20
(i) Ruby	59.34
(i) Perl	65.79
(i) Python	71.90
(i) Lua	82.91

	Mb
(c) Pascal	1.00
(c) Go	1.05
(c) C	1.17
(c) Fortran	1.24
(c) C++	1.34
(c) Ada	1.47
(c) Rust	1.54
(v) Lisp	1.92
(c) Haskell	2.45
(i) PHP	2.57
(c) Swift	2.71
(i) Python	2.80
(c) Ocaml	2.82
(v) C#	2.85
(i) Hack	3.34
(v) Racket	3.52
(i) Ruby	3.97
(c) Chapel	4.00
(v) F#	4.25
(i) JavaScript	4.59
(i) TypeScript	4.69
(v) Java	6.01
(i) Perl	6.62
(i) Lua	6.72
(v) Erlang	7.20
(i) Dart	8.64
(i) Jruby	19.84

[Programming
Language Shootout](#)

Το Πρόγραμμα Hello World

```
/* File: helloworld.c */  
  
#include <stdio.h>  
  
int main() {  
    printf("Hello world\n");  
}
```

Ας το τρέξουμε με έναν [online compiler](#)



Θέλω να εξασκηθώ - τι κάνω;

Πρακτικά προβλήματα και σε επαφή με την βιομηχανία:

- hackerrank.com
- leetcode.com

Ανταγωνιστικά / εξάσκηση σε αλγορίθμους (ή και μαθηματικά):

- adventofcode.com
- codechef.com
- codewars.com
- topcoder.com
- www.usaco.org
- projecteuler.net

Και άλλα πολλά, κάντε μια αναζήτηση για περισσότερες πηγές! (Coursera, edX, Udacity)

Για την επόμενη φορά

- Από τις σημειώσεις του κ. Σταματόπουλου μέχρι την σελίδα 19.
- Ολοκληρώστε τις εγγραφές στα εργαλεία του μαθήματος.
- Γραφτείτε σε κάποιο εργαστήριο!
- Αν θέλετε φέρτε υπολογιστή μαζί σας.
- [Programming Language](#) , [Linguistic Ambiguities](#)
- [History of C and Applications](#)

Kahoot!

Ευχαριστώ και καλή αρχή εύχομαι!
Let's Start Coding ;)