

# Προγραμματισμός I (HY120)

Διάλεξη 1:  
Εισαγωγή



# Ποιος είμαι εγώ!



2



Ναύπλιο, 4/1976-9/1993



Πάτρα, 9/1993-6/2004



Williamsburg, VA,  
USA, 7/2004-7/2006  
Χρήστος Δ. Αντωνόπουλος  
25/9/2017



Μυτιλήνη, 10/2006-2/2007



Βόλος, 2/2007 - ...

Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών  
Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

# Ο Υπεύθυνος των Εργαστηρίων

- Γιώργος Θάνος



3



Αθήνα (1976-1994)



Αθήνα (2000-2009)



Θεσσαλονίκη (1994-2000)



Βόλος (2009-σήμερα)



# Η 2η Υπεύθυνη των Εργαστηρίων

- Βάνα Ντουφεξή



4



Πάτρα (1972-1995)



Urbana-Champaign (UIUC) (1995-2000)



Chicago (Northwestern) (2000-2007)



Βόλος, 10/2007 - ...



# Ποιοι είστε εσείς;

- Υπόβαθρο;
  - Έχετε δει υπολογιστή;
  - Έχετε ανοίξει υπολογιστή;
  - Έχετε «**προγραμματίσει**» υπολογιστή;
- Τι φαντάζεστε για το μάθημα;
  - Τι περιμένετε να μάθετε;
  - Τι απαιτήσεις φαντάζεστε ότι υπάρχουν;



# Υπολογιστές, Υπολογιστές ... και Υπολογιστές



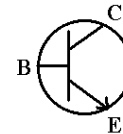
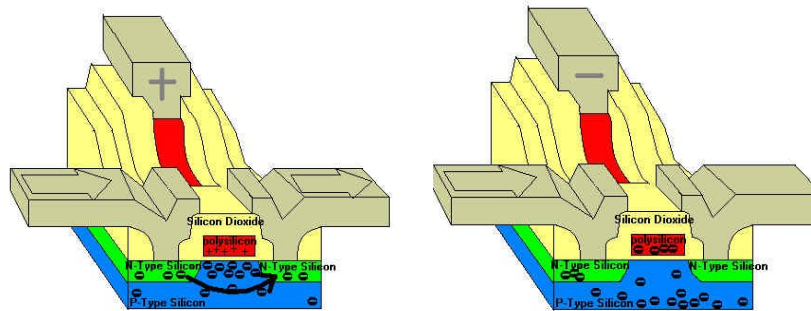
7



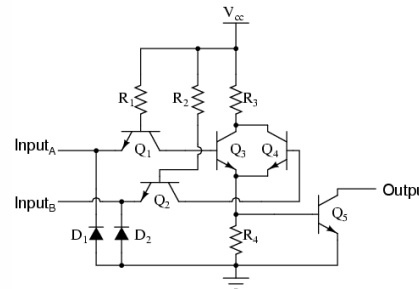
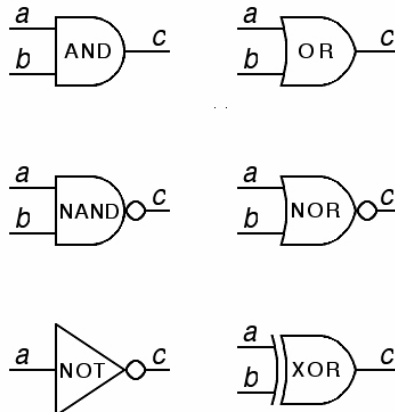
# Χτίζοντας έναν Υπολογιστή



8



Transistor



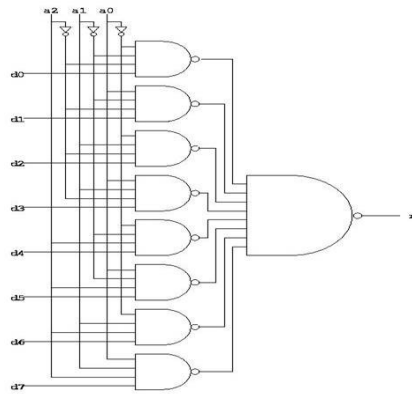
Λογικές Πύλες



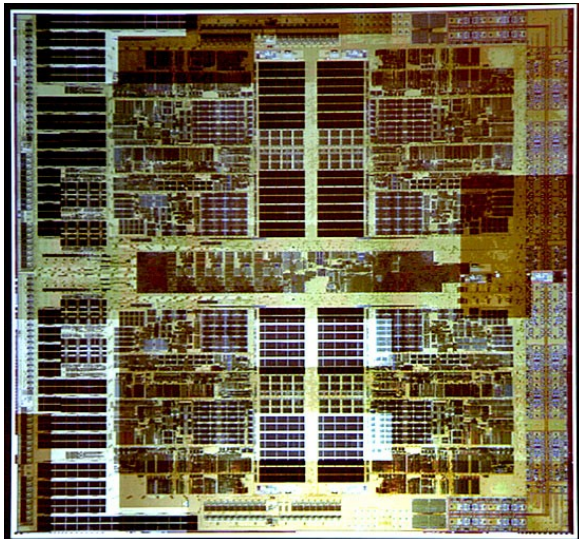
# Χτίζοντας έναν Υπολογιστή



9



Λογικά Κυκλώματα



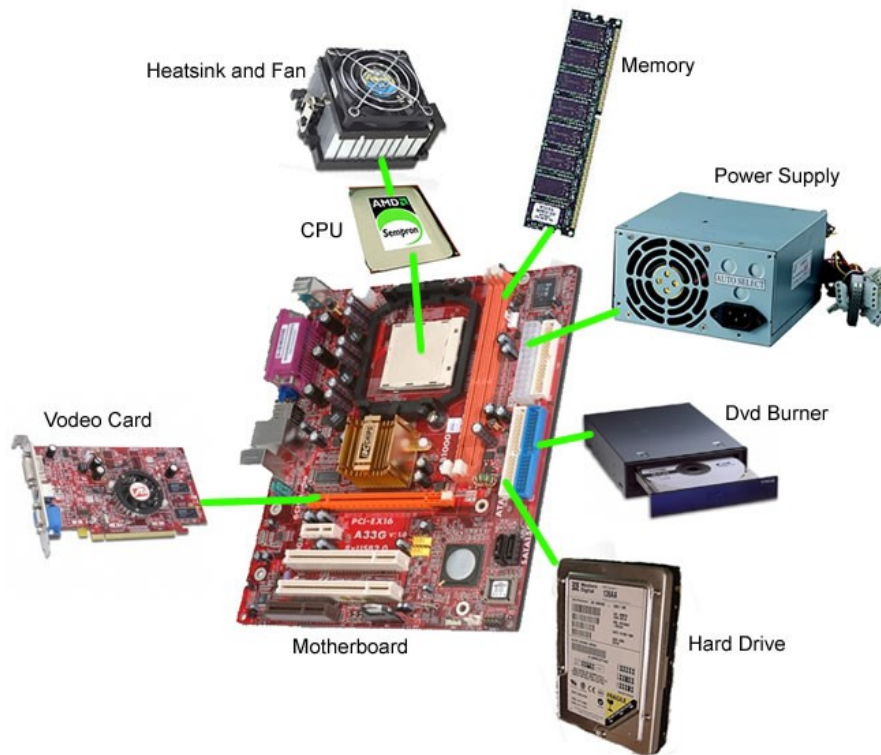
Ολοκληρωμένα  
Κυκλώματα (π.χ.  
Επεξεργαστές)



# Χτίζοντας έναν Υπολογιστή



10



## Δομικές Μονάδες - Σύστημα

# Μερικά Μυστικά των Υπολογιστών



11

- Οι υπολογιστές είναι ισχυρά εργαλεία
  - Μπορούν να κάνουν πράγματα
    - Που πριν λίγα χρόνια δεν μπορούσαμε καν να φανταστούμε
  - Μας κάνουν τη ζωή εύκολη
- Μας κάνουν τη ζωή δύσκολη
  - Οι υπολογιστές είναι **χαζοί**... Πολύ χαζοί...
    - Δεν μπορούν να κάνουν τίποτα από μόνοι τους
      - Χρειάζονται **οδηγίες** για το παραμικρό!
    - Μιλάνε μόνο μια **περίεργη γλώσσα**: 0 .. 1 .. 0 .. 0 .. 1..





12



**0 vs NULL**



# Οδηγίες στους Υπολογιστές

```

0x00007C00 33 C0 8E D0 BC 00 7C 8E C0 8E D8 BE 00 7C BF 00 3A20%{ZAZ0%{L
0x00007C10 06 B9 00 02 FC F3 A4 50 68 1C 06 CB FB B9 04 00 00 00Ph.E0!
0x00007C20 8D BE 07 80 7E 00 00 7C 0B 0F 85 10 01 83 C5 10 00%...fA
0x00007C30 E2 F1 CD 18 88 56 00 55 C6 46 11 05 C6 46 10 00 aN.L.V.U.EF..EF..
0x00007C40 B4 41 BB AA 55 CD 13 5D 72 0F 81 FB 55 AA 75 09 'A>#U|jr.üURü|
0x00007C50 F7 C1 01 00 74 03 FE 46 10 66 60 80 7E 10 00 74 +Á.tpF.F..t
0x00007C60 26 66 68 00 00 00 00 66 FF 76 08 68 00 00 68 00 &fh...fv.h.h.
0x00007C70 7C 68 01 00 68 10 00 B4 42 8A 56 00 8B F4 CD 13 |h.h..B$V..öf.
0x00007C80 9F 83 C4 10 9E EB 14 B8 01 02 8B 00 7C 8A 56 00 Yfa.äe...>|SV.
0x00007C90 8A 76 01 8A 4E 02 8A 6E 03 CD 13 66 61 73 1E FE Šv.SN.Sn.f.fas.p
0x00007CA0 4E 11 0F 85 0C 00 80 7E 00 80 0F 84 8A 00 B2 80 N.....~...$?
0x00007CB0 EB 82 55 32 E4 8A 56 00 CD 13 5D EB 9C 81 3E FE e.U2a$V.f.jeoa.>p
0x00007CC0 7D 55 AA 75 6E FF 76 00 E8 8A 00 0F 85 15 00 B0 |Urun.v.e$....."
0x00007CD0 D1 E6 64 E8 7F 00 80 DF E6 60 E8 78 00 FF E6 Næedé."Bæ'ex."æ
0x00007CE0 64 E8 71 00 88 00 8B CD 1A 66 23 C0 75 3B 66 81 deq...>f{#Au.f.
0x00007CF0 FB 54 43 50 41 75 32 81 F9 02 01 72 2C 66 68 07 üTCPAu2.ü.r.fh.
0x00007D00 BB 00 00 66 68 00 02 00 00 66 68 08 00 00 66 >.fh...fh...f
0x00007D10 53 66 53 66 55 66 68 00 00 00 66 68 00 7C 00 SFSUth...fh.|
0x00007D20 00 66 61 68 00 00 07 CD 1A 5A 32 F6 EA 00 7C 00 fah...fZ2öé.|
0x00007D30 00 CD 18 A0 87 07 EB 08 A0 B6 07 EB 03 A0 B5 07 f...é.¶|e.µ.
0x00007D40 32 E4 05 00 07 8B F0 AC 3C 00 74 FC BB 07 00 B4 2a...cö-<.tü...
0x00007D50 0E CD 10 EB F2 2B C9 E4 64 EB 00 24 02 E0 F8 24 f.eó+Éade.S.äa$
0x00007D60 02 C3 49 6E 76 61 6C 69 64 20 70 61 72 74 69 74 .Invalid partit
0x00007D70 69 6F 6E 20 74 61 62 6C 65 00 45 72 72 6F 72 20 ion table Error
0x00007D80 6C 6F 61 64 69 6E 67 20 6F 70 65 72 61 74 69 6E loading operatin
0x00007D90 67 20 73 79 73 74 65 6D 00 4D 69 73 73 69 6E 67 g system.Missing
0x00007DA0 20 6F 70 65 72 61 74 69 6E 67 20 73 79 73 74 65 operating syste
0x00007DB0 6D 00 00 00 62 7A 99 FF 21 91 4E 00 00 80 20 m...bz.!N...
0x00007DC0 21 00 07 40 3F 01 00 08 00 00 C0 46 00 00 00 !.@?....AF....
0x00007DD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

- Συμβολική γλώσσα
- Πολύ κοντά στη γλώσσα μηχανής...
- ... Αλλά κάτι μπορείς να καταλάβεις
  - Και πάλι ού μπλέξεις
  - Εκτός αν χρειαστεί ;-)

- Γλώσσα Μηχανής
- Η μητρική γλώσσα του υπολογιστή
- Ού μπλέξεις...

main:	subu	\$sp, \$sp, 32	fact:	subu	\$sp, \$sp, 32
	sw	\$ra, 20(\$sp)		sw	\$ra, 20(\$sp)
	sw	\$fp, 16(\$sp)		sw	\$fp, 16(\$sp)
	addiu	\$fp, \$sp, 28		addiu	\$fp, \$sp, 28
	li	\$v0, 4		sw	\$a0, 0(\$fp)
	la	\$a0, str		lw	\$v0, 0(\$fp)
	syscall			bgtz	\$v0, L2
	li	\$a0, 10		li	\$v0, 1
	jal	fact	L2:	j	L1
	addu	\$a0, \$v0, \$zero		lw	\$v1, 0(\$fp)
	li	\$v0, 1		subu	\$v0, \$v1, 1
	syscall			move	\$a0, \$v0
	lw	\$ra, 20(\$sp)		jal	fact
	lw	\$fp, 16(\$sp)		lw	\$v1, 0(\$fp)
	addiu	\$sp, \$sp, 32	L1:	mul	\$v0, \$v0, \$v1
	jr	\$ra		lw	\$ra, 20(\$sp)
				lw	\$fp, 16(\$sp)
				addiu	\$sp, \$sp, 32
				jr	\$ra





# Οδηγίες στους Υπολογιστές

```
static int audit_set_enabled(int state, uid_t loginuid, u32 sid)
{
    int res, rc = 0, old = audit_enabled;

    if (state < 0 || state > 2)
        return -EINVAL;

    /* check if we are locked */
    if (audit_enabled == 2)
        res = 0;
    else
        res = 1;

    if (sid) {
        char *ctx = NULL;
        u32 len;
        if ((rc = selinux_sid_to_string(sid, &ctx, &len)) == 0) {
            audit_log(NULL, GFP_KERNEL, AUDIT_CONFIG_CHANGE,
                "audit_enabled=%d old=%d by auid=%u"
                " subj=%s res=%d",
                state, old, loginuid, ctx, res);
            kfree(ctx);
        } else
            res = 0; /* Something weird, deny request */
    }
    audit_log(NULL, GFP_KERNEL, AUDIT_CONFIG_CHANGE,
        "audit_enabled=%d old=%d by auid=%u res=%d",
        state, old, loginuid, res);

    /* If we are allowed, make the change */
    if (res == 1)
        audit_enabled = state;
    /* Not allowed, update reason */
    else if (rc == 0)
        rc = -EPERM;
    return rc;
}
```

- Γλώσσες υψηλού επιπέδου
- Τώρα μάλιστα! Κάτι γίνεται!
  - Πιο κοντά στον άνθρωπο
- Ναι, αλλά ο υπολογιστής θα καταλάβει;



# Περιεχόμενο Μαθήματος

- Εισαγωγή στον προγραμματισμό.
- Το μοντέλο μνήμης και η έννοια της μεταβλητής.
- Βασικοί τύποι δεδομένων και τελεστές.
- Δομές ελέγχου, διακλαδώσεις, επαναλήψεις.
- Συναρτήσεις
  - Εμβέλεια και διάρκεια ζωής μεταβλητών
  - Κλήση συναρτήσεων
  - Πέρασμα παραμέτρων
  - Αναδρομή.
- Δείκτες.
- Πίνακες, σύνθετοι τύποι δεδομένων και δυναμικές δομές δεδομένων.
  
- **Εργαστήριο!**



# Πληροφορίες

- Βασικά συγγράμματα:
  - «**C Προγραμματισμός**», 7<sup>η</sup> έκδοση, Deitel & Deitel, Εκδόσεις Μ. Γκιούρδας.
  - «**C Από τη Θεωρία στην Εφαρμογή**», 3<sup>η</sup> έκδοση, Γ.Σ. Τσελίκης & Ν.Δ. Τσελίκας
  - «**Η γλώσσα C σε βάθος**», 5<sup>η</sup> έκδοση, Νίκος Μ. Χατζηγιαννάκης, Εκδόσεις Κλειδάριθμος ΕΠΕ
- Μπορείς να μάθεις ποδήλατο διαβάζοντας οδηγίες σε βιβλίο;
  - Ο προγραμματισμός δε μαθαίνεται με διάβασμα
  - Δουλειά στον υπολογιστή!!!
  - Πειραματισμός, λάθη, εύρεση λαθών κ.ο.κ.

# Απαιτήσεις - Αξιολόγηση



17



ΒΑΛΤΕ ΜΟΥ ΕΝΑ 5



# Απαιτήσεις – Αξιολόγηση - 1ετείς



18

- 1ετείς: Εργαστήριο + Quiz + Homeworks + Εξέταση (στο εργαστήριο)
  - Δικαίωμα εξέτασης:
    - Εργαστήριο + Quiz : Point System – Πρέπει να μαζέψετε 26 πόντους
      - Και 11,5 πόντους μετά το 6ο εργαστήριο και 2ο quiz
  - Τελικός Βαθμός:  $0.5 * \text{Εξέταση} + 0.25 * \text{Quiz} + 0.25 * \text{Homeworks}$ 
    - Για να είναι προβιβάσιμος:
      - Τελικός  $\geq 5$  (κατόπιν κλιμακώνεται σε [6, 10] )
      - Βαθμός εξέτασης  $\geq 5$

# Απαιτήσεις – Αξιολόγηση – 1ετείς – Point System



19

## ● 12 Εργαστήρια

- 0 πόντοι για απουσία / ανεπαρκή ποιότητα παραδοτέου,
- 1 πόντος για οριακά επαρκή ποιότητα παραδοτέου,
- 3 πόντοι για επαρκή / καλή ποιότητα παραδοτέου

## ● 4 Quiz

- Ζυγισμένα βάρη
- 20 πόντοι σύνολο (και στα 4)

# Απαιτήσεις – Αξιολόγηση – Μεγαλύτεροι φοιτητές



20

- Δικαίωμα εξέτασης στο εργαστήριο: Εργασία εξαμήνου  $\geq 5$ 
  - Ή επιτυχής εργασία 2016-17
- Τελικός βαθμός:
  - Βαθμός εξέτασης, κλιμακωμένος στο [6, 10]
- Επιτυχία: Τελικός βαθμός  $\geq 5$

# Απαιτήσεις – Αξιολόγηση - Όλοι



21

- Όλοι:
  - Γραπτή εξέταση (μόνο Σεπτέμβρη): Χωρίς project, εργαστήριο κλπ.
- Τελικός βαθμός:
  - Βαθμός γραπτού κλιμακωμένος στο [5, 6] (για γραπτή εξέταση)
- Επιτυχία: Τελικός βαθμός  $\geq 5$





# Ώρες Εργαστηρίου

- Θα διαλέξετε κάποιο από τα τμήματα
- Παρακολουθείτε τη σελίδα του μαθήματος και τον πίνακα ανακοινώσεων
- Αρχίζουμε **από την επόμενη εβδομάδα!!!**

# Οι καλοί λογαριασμοί...

- Δεν επιτρέπεται να μοιράζεστε κώδικα εκτός ομάδας. Το εννοούμε...



23

- Έχουμε τρόπο να βρούμε τις “εξυπνάδες”
  - Για όποιον κάνει “εξυπνάδες” (παίρνει ή δίνει κώδικα), απλά 0... Στη 2η φορά, ραντεβού του χρόνου και αναφορά στο Τμήμα...
  - Δικαιολογίες δεν ήξερα, δεν κατάλαβα δεν πιάνουν...

A screenshot of a code editor window. The main area is filled with a large block of red text, likely representing a warning or error message. The text is dense and appears to be a list of items or a detailed report. The editor's interface includes a menu bar at the top and a scroll bar on the right.A screenshot of a code editor window, similar to the one on the left. It also displays a large block of red text, which seems to be a continuation or a different part of the same report. The layout and interface elements are consistent with the first screenshot.



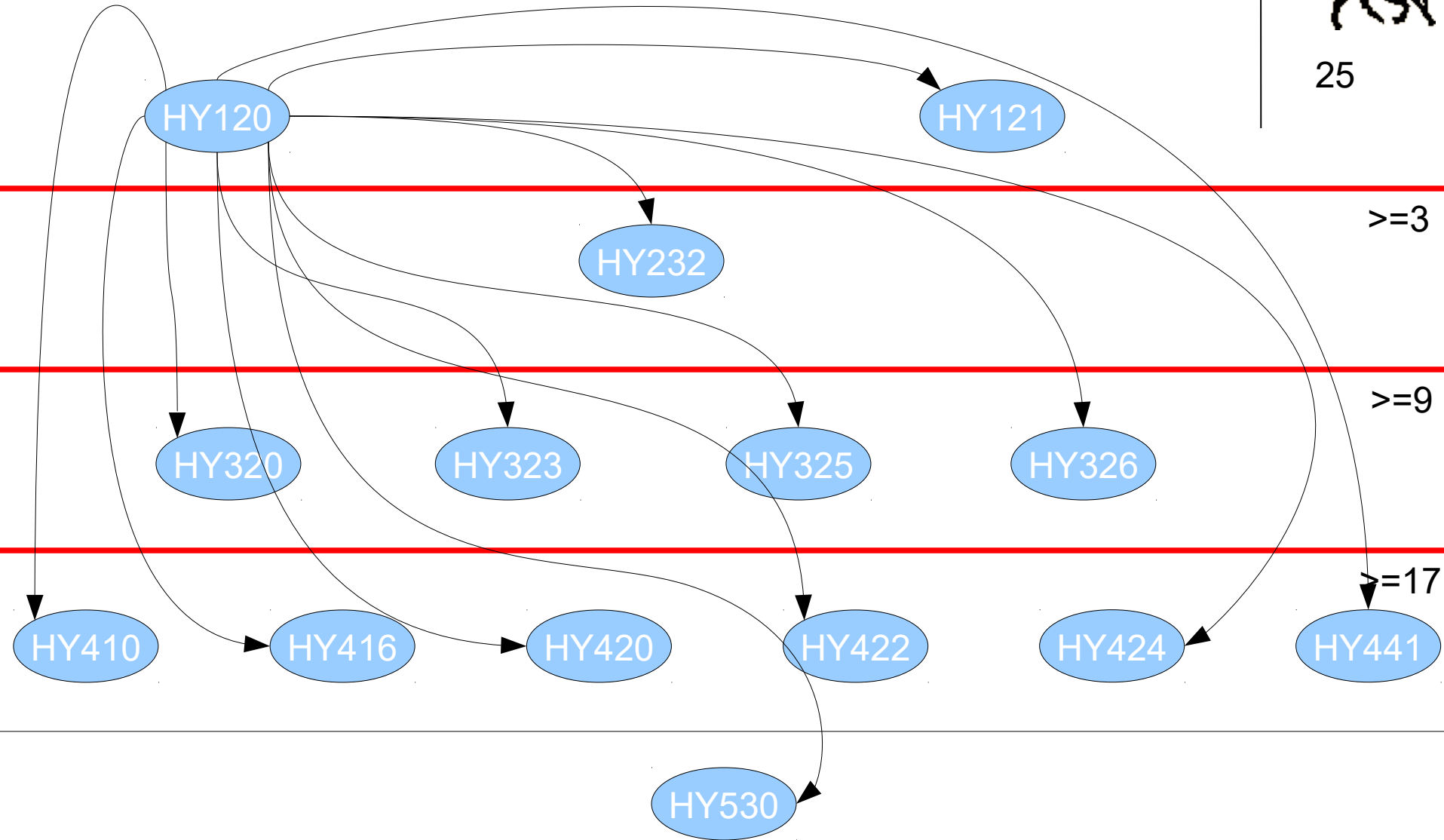
# Οι καλοί λογαριασμοί ...

- Πολιτική Τμήματος για αντιγραφές:
  - 1<sup>η</sup> φορά (σε οποιοδήποτε μάθημα): Απώλεια μαθήματος για 1 έτος
  - 2<sup>η</sup> φορά (σε οποιοδήποτε μάθημα): Απώλεια εξαμήνου
  - 3<sup>η</sup> φορά (σε οποιοδήποτε μάθημα): Απώλεια έτους και αναφορά στη Σύγκλητο

# Πρόγραμμα Σπουδών



25



# Και γιατί να προσπαθήσω;

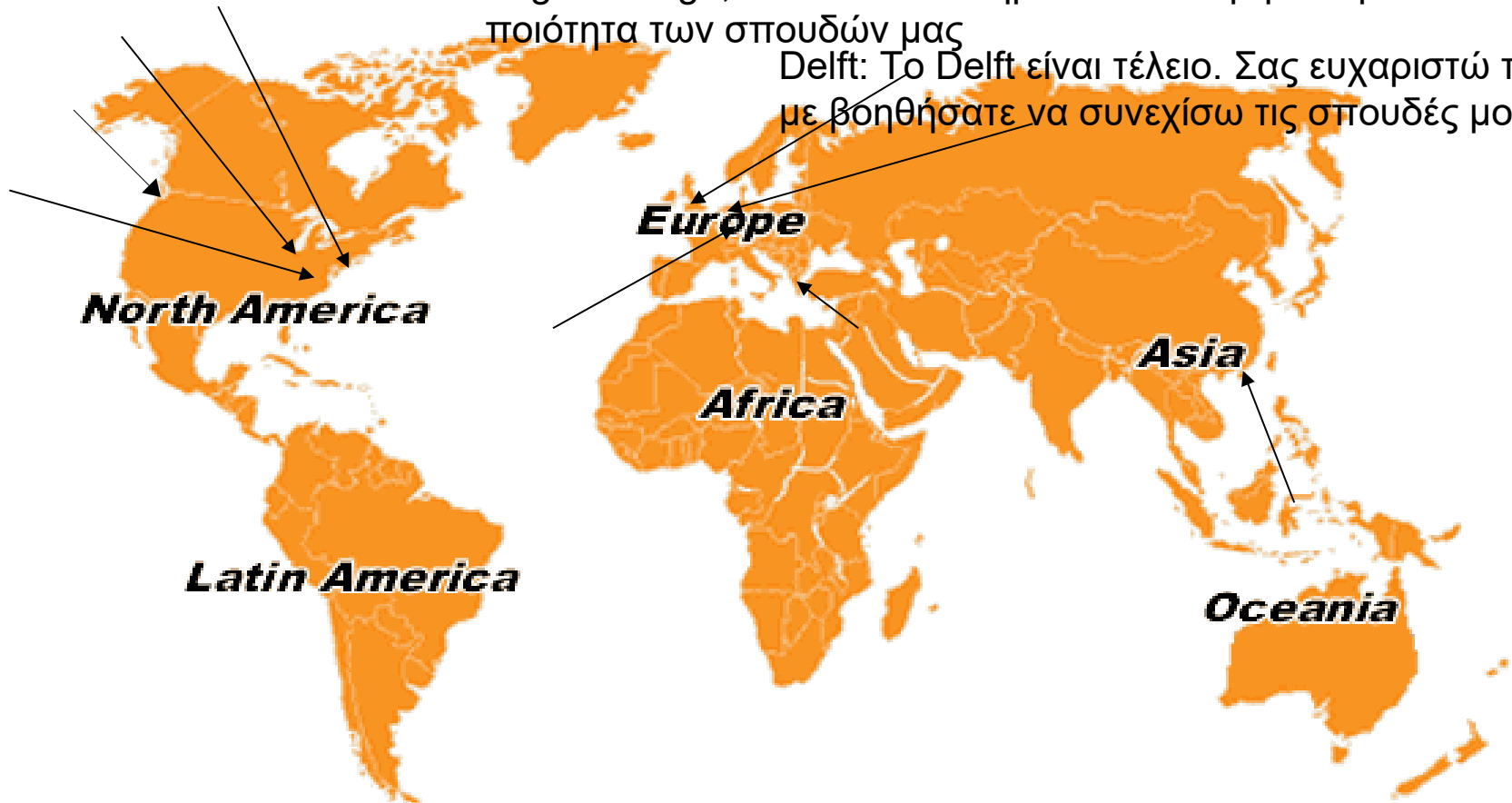


26

NY University: Εδώ στο Πανεπιστήμιο πάντως τα πράγματα πολύ καλά.  
Δε θέλω να πω μεγάλες κουβέντες, αλλά νομίζω ότι άμα έχεις τελειώσει  
το Τμήμα το επίπεδό σου είναι πολύ καλό. Το ίδιο και η ικανότητά σου  
να αντιμετωπίζεις τις δυσκολίες.

King's College, London: Όταν ήρθα εδώ εκτίμησα την  
ποιότητα των σπουδών μας

Delft: Το Delft είναι τέλειο. Σας ευχαριστώ που  
με βοηθήσατε να συνεχίσω τις σπουδές μου εκεί





- Ιστοσελίδα + forum μαθήματος
  - <http://eclass.uth.gr/eclass/courses/MHX367/>
- Λίστα ηλεκτρονικού ταχυδρομείου
  - <http://inf-server.inf.uth.gr/mailman/listinfo/ce120>





- ΕΠΙΚΟΙΝΩΝΙΑ με τον διδάσκοντα:
  - Γκλαβάνη 37, Γραφείο Β3/5
  - Ώρες γραφείου: στη σελίδα του μαθήματος.
  - E-mail: [cdantonop@gmail.com](mailto:cdantonop@gmail.com), [cda@inf.uth.gr](mailto:cda@inf.uth.gr)
  
- ΕΠΙΚΟΙΝΩΝΙΑ με τον υπεύθυνο εργαστηρίων:
  - Γκλαβάνη 37, Γραφείο Γ5/8
  - E-mail: [gthanos@uth.gr](mailto:gthanos@uth.gr)
  
- ΕΠΙΚΟΙΝΩΝΙΑ με την υπεύθυνη εργαστηρίων:
  - Γκλαβάνη 37, Γραφείο Γ5/10
  - E-mail: [doufexi@gmail.com](mailto:doufexi@gmail.com), [vdoufexi@inf.uth.gr](mailto:vdoufexi@inf.uth.gr)

# Μεθοδολογία Επίλυσης Προβλήματος



29

- **Ανάλυση** του προβλήματος
  - Τι έχουμε;
  - Τι ακριβώς θέλουμε να βρούμε;
  - Πώς σπάει σε μικρότερα / απλούστερα προβλήματα;
  - Χονδρικά ποια στρατηγική πρέπει να ακολουθήσουμε;
- **Αλγόριθμος**
  - Ακριβής ακολουθία πεπερασμένων βημάτων που πρέπει να ακολουθηθούν για να λυθεί το πρόβλημα
  - Απαιτήσεις
    - Ορθότητα
    - Καλή επίδοση
    - «Οικονομία» πόρων
  - Για το ίδιο πρόβλημα πιθανόν να υπάρχουν  $>1$  κατάλληλοι αλγόριθμοι

# Μεθοδολογία Επίλυσης Προβλήματος

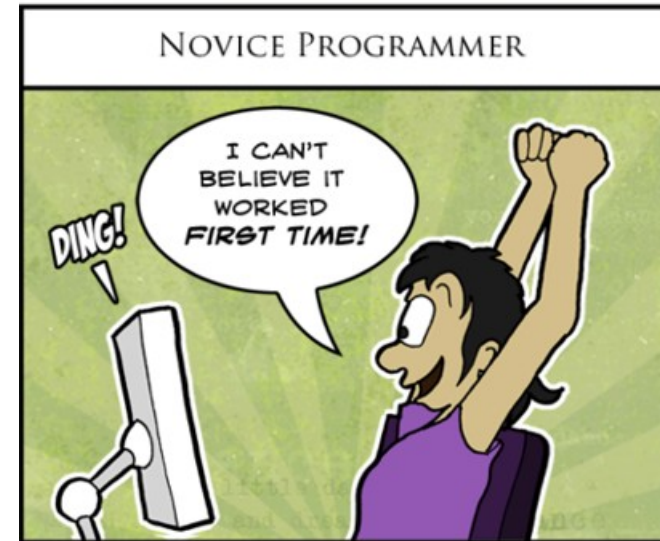


- **Κωδικοποίηση**

- «Μετάφραση» του αλγόριθμου σε γλώσσα που καταλαβαίνει ο υπολογιστής
  - C στην περίπτωση μας

- **Δοκιμή και Αποσφαλμάτωση**

- Ουδείς αλάνθαστος!
  - Το πιο «σπαστικό» κομμάτι...
  - ...Και το πιο χρονοβόρο
  - ...Και το πιο σημαντικό



# Τι Είναι Πρόγραμμα και Γλώσσα Προγραμματισμού;



31

- **Πρόγραμμα**: Αλγόριθμος γραμμένος σε μια γλώσσα προγραμματισμού
  - Αυτή που «καταλαβαίνει» ο υπολογιστής
    - Με λίγη βοήθεια... Ο υπολογιστής ξέρει μόνο από 0 και 1
- **Γλώσσα προγραμματισμού**
  - Σαν τις κανονικές γλώσσες
    - Μόνο πιο σαφής και τυπική ...
  - Έχει **λεξιλόγιο / σημασιολογία**
  - Έχει **συντακτικό**

# ΣΥΝΤΑΚΤΙΚΟ



- Το συντακτικό ορίζει το σύνολο των επιτρεπτών προτάσεων (ακολουθία συμβόλων).
- Επιτρεπτή πρόταση = συντακτικά ορθό πρόγραμμα
  - Μπορεί να μεταφραστεί σε γλώσσα μηχανής και να εκτελεστεί.
- Ειδικές «γλώσσες» περιγραφής των κανόνων σύνταξης γλωσσών προγραμματισμού
  - Extended Backus Naur Form (EBNF)
  - Συντακτικά διαγράμματα



# Σύνταξη και Σημασία

- **Σύνταξη:** κανόνες για τον σχηματισμό προτάσεων.
- **Σημασία:** το νόημα που έχουν οι συντακτικά επιτρεπτές προτάσεις.
  - Η σύνταξη μιας γλώσσας δεν μπορεί να ορίσει ταυτόχρονα και την σημασία της.
  - Υπάρχουν όμως κανόνες!
- Υπάρχουν συντακτικά επιτρεπτές προτάσεις που δεν έχουν ορισμένη (μια μοναδική) σημασία;
  - Στις ανθρώπινες γλώσσες: Ναι
    - Μαζί μιλάμε και χώρια καταλαβαίνομαστε
  - Σε γλώσσες προγραμματισμού: Κατά κανόνα όχι





# Για παράδειγμα ...

- Συντακτικά ορθές προτάσεις:

$$-2+13$$

$$15/5-2$$

- Όμως, σε τι τιμές αντιστοιχούν;

$$-2+13 : \quad (-2) + (13) \quad \text{ή} \quad -(2+13)$$

$$15/5-2 : \quad (15/5) - 2 \quad \text{ή} \quad 15 / (5-2)$$



# Εκτέλεση Προγράμματος

επέλεξε την πρώτη εντολή

εκτέλεσε την εντολή

επέλεξε την «επόμενη»  
εντολή

μνήμη

διάβασμα

γράψιμο

μεταβλητές  
προγράμματος

Το σημειωματάριο  
του προγράμματος...



# Έλεγχος Ροής Εκτέλεσης

- Ο προγραμματιστής μπορεί να επηρεάσει την ροή εκτέλεσης του προγράμματος μέσω **εντολών ελέγχου**.
  - Εντολές ελέγχου: διαβάζουν την κατάσταση του προγράμματος και ανάλογα μεταφέρουν την εκτέλεση σε ένα διαφορετικό σημείο του κώδικα.
  - Π.χ `if <έκφραση> goto <n>`
    - Αν η <έκφραση> είναι αληθής τότε η εκτέλεση μεταφέρεται στην εντολή με αριθμό <n>.
- Μπορούμε εύκολα να φτιάξουμε προγράμματα με εντολές που εκτελούνται **υπό συνθήκη** ή/και **επανειλημμένα**.



# Για Παράδειγμα...

παράμετροι

πρόγραμμα P(x, y)

```
1:s = 0
2:if (x==0) goto 6
3:s = s + y
4:x = x -1
5:if (1==1) goto 2
6:print s
```

εκτέλεση P(0, 5)

```
s = 0
if (x==0) goto 6
print s
```

εκτέλεση P(3, 5)

```
s = 0
if (x==0) goto 6
s = s + y
x = x -1
if (1==1) goto 2
if (x==0) goto 6
s = s + y
x = x -1
if (1==1) goto 2
if (x==0) goto 6
s = s + y
x = x -1
if (1==1) goto 2
if (x==0) goto 6
print s
```



# Υποπρογράμματα

- Ομάδα εντολών (και δεδομένων) που είναι **συντακτικά** ή/και **εκτελεστικά ανεξάρτητη** (αυτόνομη) από τον υπόλοιπο κώδικα.
  - Ένα πρόγραμμα μπορεί να αποτελείται από (ή να χρησιμοποιεί) πολλά διαφορετικά υποπρογράμματα.
  - Όταν καλείται ένα υποπρόγραμμα, η εκτέλεση **μεταφέρεται** στο υποπρόγραμμα
    - Όταν τερματιστεί η εκτέλεση του, **επιστρέφει** μαγικά στο πρόγραμμα που πραγματοποίησε την κλήση.
  - Να τα χρησιμοποιείτε... please!!!



# Εκτέλεση με Υποπρογράμματα

