

# ΕΠΛ 033: ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΓΙΑ ΜΗΧΑΝΙΚΟΥΣ

Μάριος Belk, Τμήμα Πληροφορικής, Πανεπιστήμιο Κύπρου

Email: [belk@cs.ucy.ac.cy](mailto:belk@cs.ucy.ac.cy)



# Γενικές Πληροφορίες

1

<b>Διδάσκων Καθηγητής:</b>	Μάριος Belk
<b>Γραφείο:</b>	ΘΕΕ01 – 211 (2 <sup>ος</sup> όροφος, Πτέρυγα Γ')
<b>Τηλέφωνο:</b>	22 892685
<b>E-mail:</b>	<a href="mailto:belk@cs.ucy.ac.cy">belk@cs.ucy.ac.cy</a>
<b>Ιστοσελίδα Μαθήματος:</b>	blackboard.cs.ucy.ac.cy → ΕΠΛ033
<b>Διαλέξεις:</b>	Δευτέρα & Πέμπτη, 10:30 -12:00
<b>Αίθουσα:</b>	ΧΩΔ-01 Αίθουσα 102 (Πανεπιστημιούπολη)
<b>Φροντιστήρια:</b>	
<b>Ώρες γραφείου:</b>	Δευτέρα 14:00 – 16:00 ή μετά από συνεννόηση
<b>Εργαστήρια:</b>	1Α: Τετάρτη, 16:00-18:00 (Αίθ. Β103 – ΘΕΕ01) 1Β: Τετάρτη, 14:00 - 16:00, (Αίθ. Β103 – ΘΕΕ01)
<b>Υπεύθυνος Εργαστηρίου:</b>	Νάταλη Τεμενέ

# Σελίδα Μαθήματος

2

[https://www.cs.ucy.ac.cy/courses/  
EPL033](https://www.cs.ucy.ac.cy/courses/EPL033)

- Πληροφορίες
- Διαφάνειες
- Παράδοση εργασιών
- Βαθμολογία
- Ανακοινώσεις
- Επικοινωνία

# Τι θα μάθουμε;

3

- Κατανόηση της έννοιας του προγραμματισμού
- Θεμελίωση της αλγοριθμικής σκέψης για την επίλυση προβλημάτων με τον προγραμματισμό
- Ικανότητα σχεδιασμού, υλοποίησης, δοκιμής, αποσφαλμάτωσης και τεκμηρίωσης προγραμμάτων στη γλώσσα προγραμματισμού C

# Τι θα μάθουμε;

4

- Θα ξέρουμε πώς να αξιοποιήσουμε τον προγραμματισμό για **ευκολία** και **απόδοση** στη δουλειά μας και στην καθημερινότητά μας
- Οι πλείστες εφαρμογές επιτρέπουν τον μικροπρογραμματισμό τους ώστε ο χρήστης να αυτοματοποιήσει επαναλαμβανόμενες διαδικασίες (Word, Excell, AutoCad, 3DsMax, Λογιστικά προγράμματα, κ.τ.λ.)

# Γιατί να το μάθουμε;

5

- Θα γίνει εφικτή η **επικοινωνία** μας με κάθε είδους προγραμματιστές
- Θα μπορούμε πολύ εύκολα να **κατανοήσουμε κώδικα** προγραμμάτων και να κάνουμε αλλαγές
- Θα έχουμε μεγάλο **πλεονέκτημα** στην αγορά εργασίας
- Θα καλλιεργήσουμε την αλγοριθμική σκέψη, όντας ένα πολύ δυνατό **εργαλείο** στη **καθημερινότητα**
- Θα μπορούμε να αξιοποιήσουμε **πλήρως** τους υπολογιστές και την τεχνολογία

# Πως θα το μάθουμε;

6

- **Διαλέξεις**
  - Υποχρεωτική παρακολούθηση
  - Παρουσιάζονται οι διάφορες έννοιες
  - Επιδιώκεται ο διάλογος και η ανταλλαγή επιχειρημάτων
- **Ερωτήσεις Ελέγχου**
  - Προσωπικός έλεγχος εάν κατανοήσατε τις έννοιες των διαλέξεων
  - Να τις απαντάτε μόνοι σας ΠΡΙΝ ΤΗΝ ΕΠΟΜΕΝΗ ΔΙΑΛΕΞΗ
  - Θα είναι πολύ πιθανόν μέρος των πολλαπλών ερωτήσεων στις εξετάσεις
- **Εργαστήρια**
  - Υποχρεωτική παρακολούθηση
  - Παρουσίαση εργαλείων
  - Εμπέδωση εννοιών σε καθαρά πρακτικό επίπεδο
- **Ασκήσεις και πρακτική εξάσκηση (x4)**
  - Υποχρεωτική συμμετοχή
  - Εφαρμογή εννοιών προγραμματισμού για την επίλυση καθημερινών προβλημάτων σε γλώσσα προγραμματισμού C

# Πως θα το μάθουμε;

7

## □ Ενδιάμεσες Εξετάσεις (quiz x2)

### □ Υποχρεωτική συμμετοχή

#### □ Quiz 1

- Γραπτή Ενδιάμεση
- Εργαστηριακή Ενδιάμεση
- Βασικές έννοιες

#### □ Quiz 2

- Γραπτή Ενδιάμεση
- Προγραμματισμός

## □ Τελική Εξετάση

### □ Υποχρεωτική συμμετοχή

### □ Καλός Προγραμματισμός



# Αξιολόγηση και Βαθμολόγηση

Κατ' οίκον εργασίες και Διαγνωστικά	25%
Ενδιάμεση Γραπτή Εξέταση	25%
Τελική Γραπτή Εξέταση	50%

Η εκπόνηση όλων των εργασιών καθώς και η συμμετοχή σε όλες τις εξετάσεις και διαγνωστικά είναι **υποχρεωτική**

*Απαραίτητες προϋποθέσεις επιτυχίας στο μάθημα είναι:*

- Η **συμμετοχή** του φοιτητή και στις 3 εξετάσεις (ενδιάμεσες και τελική εξέταση)
- Η **παράδοση** όλων των εργαστηριακών ασκήσεων και διαγνωστικών

# Τεκμηρίωση Πιστωτικών Μονάδων

Ώρες εργασίας του μέσου φοιτητή/φοιτήτριας για την επίτευξη μαθησιακών αποτελεσμάτων

**ΕΠΛ033 = 5 ΠΜ = 150-180 ώρες εργασίας**

- |   |                 |
|---|-----------------|
| □ Παρακολούθηση διαλέξεων και εργαστηρίων:<br>(6 ώρες x 11 εβδομάδες) | 66 ώρες         |
| □ Μελέτη ερωτήσεων ελέγχου:<br>(1 ώρα μετά από κάθε διάλεξη)          | 24 ώρες         |
| □ Εκπόνηση προγραμματιστικών εργασιών:<br>(10-20 ώρες ανά εργασία)    | 45 ώρες         |
| □ Μελέτη για τις εξετάσεις:<br>(10-20 ώρες πριν από κάθε εξέταση)     | 45 ώρες         |
| □ <b>Σύνολο ωρών:</b>   | <b>210 ώρες</b> |
-

# Βιβλιογραφία

- J.R. Hanly και E.B. Koffman, Problem Solving & Program Design in C, 6η έκδοση, Addison-Wesley, 2010.
- Α.Δ. Καμέας, Τεχνικές Προγραμματισμού, Τόμος Β της Θεματικής Ενότητας Εισαγωγή στην Πληροφορική, Ελληνικό Ανοικτό Πανεπιστήμιο, 2000.
- Κ. Θραμπουλίδης, Γλώσσες Προγραμματισμού, Τόμος Δ της Θεματικής Ενότητας Εισαγωγή στην Πληροφορική, Ελληνικό Ανοικτό Πανεπιστήμιο, 2000.



# Υπολογιστές και Προγραμματισμός

# Αναπαράσταση Πληροφοριών

12

- Με τη χρήση «ψηφιακών» ηλεκτρονικών κυκλωμάτων
- «Ψηφιακά» είναι τα ηλεκτρονικά κυκλώματα που μπορούν να βρίσκονται μόνο σε διακριτές καταστάσεις
  - ▣ κάθε κατάσταση αντιπροσωπεύει ένα ψηφίο
- Ο Η/Υ βασίζεται σε ψηφιακά κυκλώματα που μπορούν να βρίσκονται σε *μια* από **δύο** καταστάσεις:
  - ▣ Ανοιχτό (αγωγή ρεύματος)     **1**
  - ▣ Κλειστό (διακοπή ρεύματος)   **0**

# Αναπαράσταση Πληροφοριών

13

- Κάθε πληροφορία μετατρέπεται σε μια ακολουθία από ψηφία 0 και 1
  - Δεκαδικοί αριθμοί (μετατρέπονται σε δυαδικούς)
  - Γράμματα και χαρακτήρες (μετατρέπονται με βάση κοινούς πίνακες αντιστοίχισης)
  - Εικόνες
  - Ήχος
  - Εντολές (με βάση προκαθορισμένες αντιστοιχίσεις από τον κατασκευαστή)

# Ψηφιακή πληροφορία

14

- Ποια είναι η μικρότερη μονάδα πληροφορίας;
  - ▣ Το δυαδικό ψηφίο (= δύο πιθανές καταστάσεις)
- Πώς ονομάζεται το δυαδικό ψηφίο;
  - ▣ bit (**b**inary **d**igit)

# Ομαδοποίηση ακολουθιών bits

15

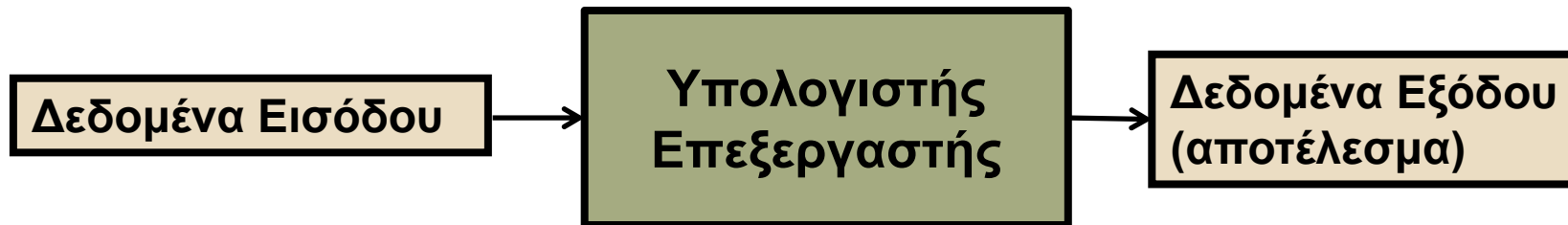
- bit – b                    ένα δυαδικό ψηφίο (0 ή 1)
- byte – B                     $2^3$  bits (π.χ. 10010010)
- (word                    4 bytes)
- KiloByte – KB             $2^{10}$  bytes
- MegaByte – MB            $2^{20}$  bytes (1024x1024)
- GigaByte – GB             $2^{30}$  bytes (1024x1024x1024)
- TeraByte – TB             $2^{40}$  bytes



# Ο Υπολογιστής

16

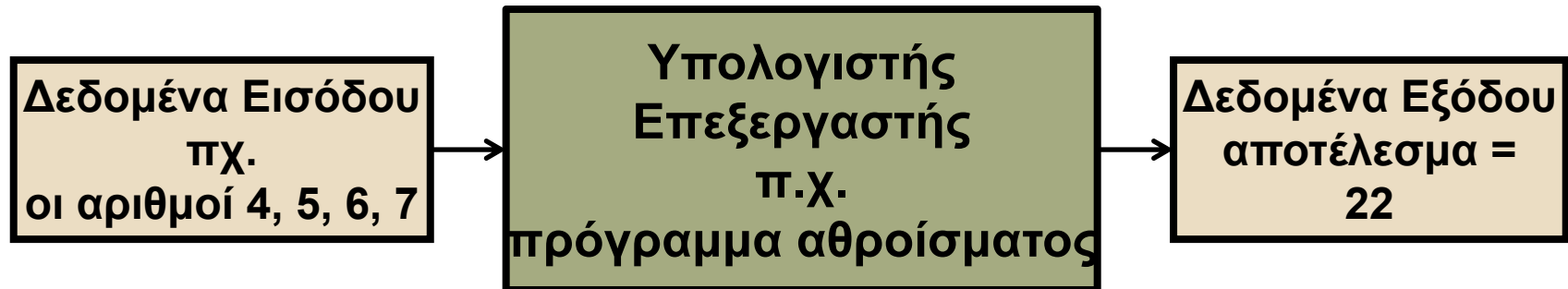
- Η κατάλληλη διασύνδεση ψηφιακών κυκλωμάτων επιτρέπει την αποθήκευση και επεξεργασία δεδομένων σε ψηφιακή μορφή



# Ο Υπολογιστής

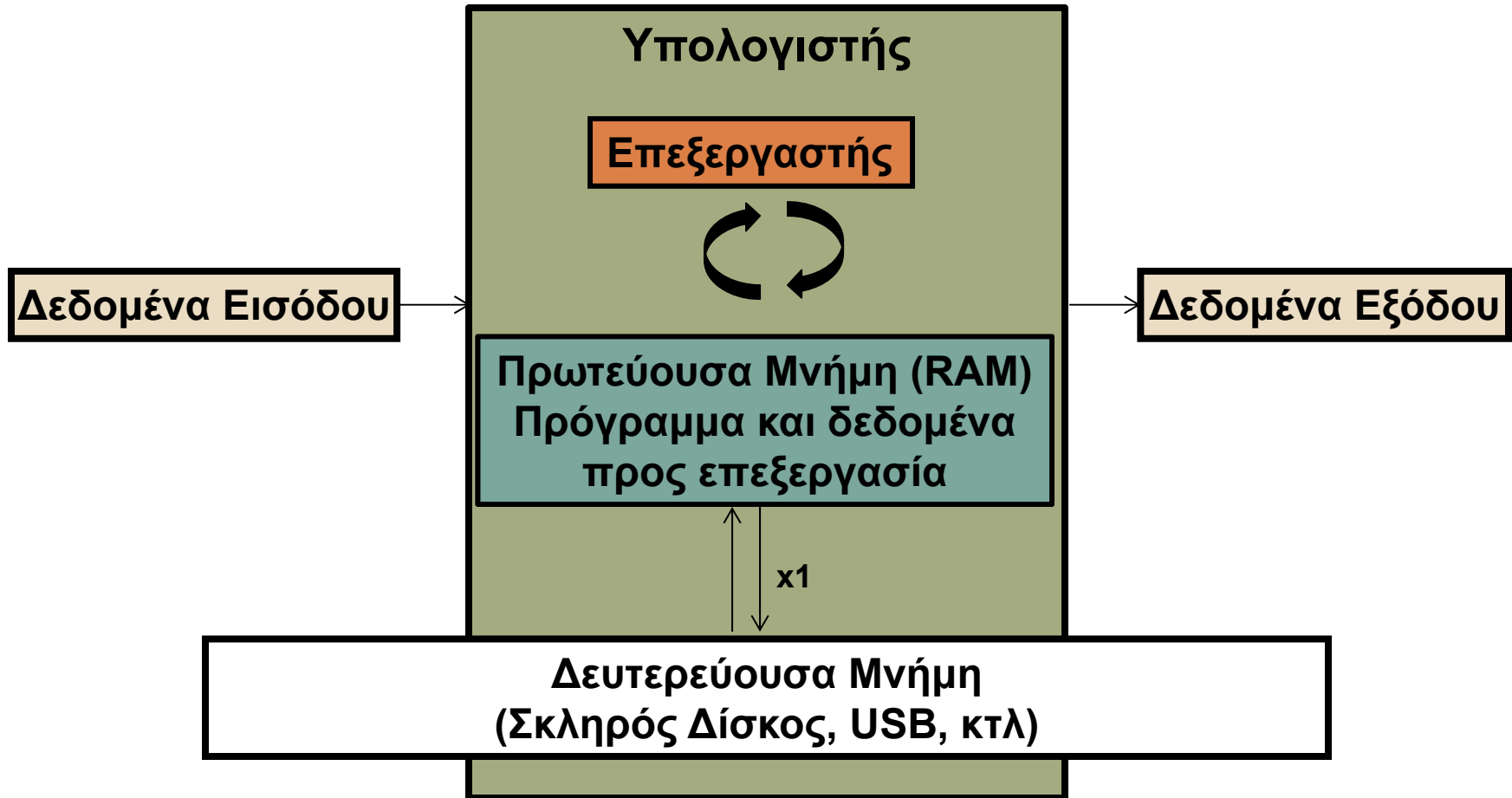
17

- Η κατάλληλη διασύνδεση ψηφιακών κυκλωμάτων επιτρέπει την αποθήκευση και επεξεργασία δεδομένων σε ψηφιακή μορφή



# Ο Υπολογιστής

18



# Ο Υπολογιστής

19

- Επεξεργαστής
  - ▣ Εκτελεί εντολές πάνω σε δεδομένα
  - ▣ Αυτά τα παίρνει απευθείας από την RAM
  - ▣ Τα αποτελέσματα τα στέλνει πίσω στην RAM
- Πρωτεύουσα μνήμη (RAM)
  - ▣ Κρατάει προγράμματα και δεδομένα προς επεξεργασία μέχρι να τελειώσουν ή να σβήσει ο Η/Υ
- Δευτερεύουσα μνήμη
  - ▣ Αποθηκεύει αρχεία

# Ο Υπολογιστής

## Κύκλος εκτέλεσης προγράμματος

1. Ο χρήστης δίνει την εντολή να τρέξει ένα πρόγραμμα
2. Ο Επεξεργαστής φορτώνει το πρόγραμμα στην RAM από το αρχείο που βρίσκεται στη Δευτερεύουσα Μνήμη
3. Ο Επεξεργαστής διαβάζει την πρώτη/επόμενη εντολή του προγράμματος αυτού από την RAM και την εκτελεί
  - Μπορεί να ζητάει δεδομένα εισόδου από τον χρήστη
  - Μπορεί να επεξεργάζεται κάποια δεδομένα που δοθήκαν ήδη ή δημιουργηθήκαν από το ίδιο το πρόγραμμα και υπάρχουν στην RAM
  - Μπορεί να παράγει κάποια (ενδιάμεσα) δεδομένα/αποτελέσματα και τα αποθηκεύει στη RAM ή τα παρουσιάζει/παραδίνει στον χρήστη

# Ο Υπολογιστής

21

- Επεξεργαστής
  - ▣ Εγκέφαλος
- Πρωτεύουσα μνήμη (RAM)
  - ▣ Χρησιμοποιείται σαν το Γραφείο εργασίας
  - ▣ Μοιάζει σαν ένα μεγάλο σύνολο από μικρές θυρίδες ίδου μεγέθους
  - ▣ Κάθε θυρίδα περιέχει μια ομάδα από bits (λέξη του υπολογιστή)
  - ▣ Κάθε θυρίδα έχει την δική της διεύθυνση (αριθμό)
- Δευτερεύουσα μνήμη
  - ▣ Χρησιμοποιείται σαν Αρχεία οργανωμένα σε συρτάρια
  - ▣ Μοιάζει με τεράστια αποθήκη με παλέτες που κουβαλάν αρχεία

35

**Free Editing Software**  
Download Now >

Download free audio editor program. Easy audio editor everyone can use.

Go Parallel  
Translating Multicore Power into Application Performance  
intel slashdotmedia

DEV **Orwell Dev-C++**  
A portable C/C++/C++11 IDE  
Brought to you by orwelldevcpp

Summary Files Reviews Support External Link Tracker Code Forums

★ 4.8 Stars (352)  
↓ 24.346 Downloads (This Week)  
Last Update: 6 days ago

Download  
Dev-Cpp 5.5.3 TDM-GCC x84 4.7.1 Setup.exe

Browse All Files



Description  
A maintained version of Dev-C++  
[Orwell Dev-C++ Web Site >](#)

Categories License  
Integrated Development Environments (IDE) GNU General Public License version 3.0 (GPLv3)

Go Parallel  
Translating Multicore Power into Application Performance

Stay connected, up-to-date, and informed on all things parallel development via Go Parallel, where you'll find viewpoints, how-to's, software tools, and educational information to help your software development work shine.

Most Recent Posts:

- National Renewable Energy Lab Pic...
- HTML5 Advances into Parallelism
- Understanding Xeon Phi for Progra...
- Database-Aided Parallel Processing?  
An interesting debate on Slashdot tries to untangl...

# Ερωτήσεις ελέγχου

23

## 1. Αντιστοιχήστε τα εργαλεία του μαθήματος με τον σκοπό τους:

Διαλέξεις	Εμπέδωση και αξιολόγηση φοιτητών και εντοπισμός αδυναμιών
Ερωτήσεις ελέγχου	Εμπέδωση και αξιολόγηση γνώσεων που αποκτήθηκαν
Εργαστήρια	Έλεγχος κατανόησης εννοιών
Εργασίες	Κατανόηση εννοιών
Quiz	Εκμάθηση προγραμματισμού
Τελική	Πρακτική εκμάθηση προγραμματισμού και ασκήσεις που βοηθάνε στην εκπόνηση εργασιών



# Ερωτήσεις ελέγχου

24

2. Πως αναπαρησάτε ένας δεκαδικός αριθμός στον Η/Υ;
3. Πως αναπαρησάτε ένα γράμμα στον Η/Υ;
4. Ποια είναι τα κύρια τμήματα ενός Η/Υ;
5. Ποιος είναι ο κύκλος εκτέλεσης ενός προγράμματος;