

# Εισαγωγή στον Προγραμματισμό

## ΔΙΑΔΙΚΑΣΤΙΚΑ

Δεν παραδίδονται σετ ασκήσεων τις 3 πρώτες εβδομάδες διδασκαλίας. Κατόπιν ισχύει ότι σας πει ο αντίστοιχος διδάσκων.

Στις εξετάσεις υπάρχει 1 θέμα στη θεωρία από τις 3 πρώτες εβδομάδες

Σημείωση: Η αντιγραφή ασκήσεων/θεμάτων μηδενίζει όλα τα αντίγραφα και το πρωτότυπο.

Στις παραδόσεις παίρνουμε παρουσίες

Παναγιώτης Τζουνάκης



# Εισαγωγή στον Προγραμματισμό ΛΕΙΤΟΥΡΓΙΚΑ

Απαραίτητος ο ιδρυματικός λογαριασμός ΑΠΘ (βλ. <https://it.auth.gr/el/account/univID> )



# Εισαγωγή στον Προγραμματισμό

## ΛΕΙΤΟΥΡΓΙΚΑ

- 36 Η/Υ με Λειτουργικό Σύστημα MS Windows 10
- Δεν γίνεται backup στα αρχεία χρηστών. Οι φοιτητές διατηρούν τα αρχεία τους σε δικό τους αποθηκευτικό μέσο (usb disk, <https://it.auth.gr/el/cloudServices/myfiles>, Google drive, κλπ., κλπ. )
- Στο τέλος του μαθήματος κάνουμε Shutdown τα MS Windows

Παναγιώτης Τζουνάκης



# Εισαγωγή στον Προγραμματισμό

## ΛΕΙΤΟΥΡΓΙΚΑ

Ανάρτηση σημειώσεων (θεωρία) στον ιστοχώρο του εργαστηρίου Η/Υ : <https://clab.math.auth.gr/el/progspr22>



Εργαστήριο Η/Υ

Τμήμα Επιστήμης Υπολογιστών και Αριθμητικής Ανάλυσης, Τμήμα Μαθηματικών,  
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Αρχική > "Εισαγωγή στον Προγραμματισμό" την Άνοιξη του 2022

### "Εισαγωγή στον Προγραμματισμό" την Άνοιξη του 2022

Προβλή Επεξεργασία

Posted By [rij@math.auth.gr](mailto:rij@math.auth.gr) Στις Saturday, 26 February 2022

Βρίσκεστε στον προσωρινό τόπο ανάρτησης εκπαιδευτικό κ.α. ηλεκτρονικού υλικού για το Εισαγωγικό Μέρος (θεωρία) του εργαστηριακού μαθήματος "Εισαγωγή στον Προγραμματισμό" την Άνοιξη του 2022.

#### ΕΞΕΤΑΣΗ ΜΑΘΗΜΑΤΟΣ

Λόγω κορονοϊού, λεπτομέρειες για την εξέταση του μαθήματος θα ανακοινωθούν έγκαιρα, σε εύθετο χρόνο, αναλόγως με τις αποφάσεις και οδηγίες των αρμοδίων οργάνων του ΑΠΘ.

ΩΡΕΣ ΓΡΑΦΕΙΟΥ τις εβδομάδες των παραδόσεων (στο εργαστήριο Η/Υ) : Δευτέρα 15:00 - 16:30 και Παρασκευή 12:00 - 16:00

**ΗΛΕΚΤΡΟΝΙΚΕΣ ΩΡΕΣ ΓΡΑΦΕΙΟΥ** : Όσο το εργαστήριο Η/Υ είναι κλειστό, όποιος φοιτητής επιθυμεί να επικοινωνήσει σε ώρα γραφείου, στέλνει email ή AT math DOT auth DOT gr , με προτεινόμενη μέρα και ώρα, κλείνουμε ραντεβού και συνομιλούμε μέσω Skype ή zoom.

Επίσης μπορείτε πάντοτε να στέλνετε email messages με απορίες κλιτ στην ίδια διεύθυνση.

Το υφολόγιο πρόγραμμα μαθημάτων του τρέχοντος εξάμηνου τηρείται στην ιστοσελίδα <https://drive.google.com/file/d/1f60x4b-PhAZHEi9vD3pe774xvN8vbnw/view> σύμφωνα με τον πρόσφατο "Κανονισμό Λειτουργίας". Σ.Θ.Ε. <http://www.scl.auth.gr/sites/default/files/%CE%93%CE%95%CE%9D%CE%99%CE%9A...> οι διήμερες παραδόσεις του μαθήματος θα διεξάγονται ως συνεχόμενο εναετητάλετο χωρίς ενδιάμεσο διάλειμμα (ένταξη "και τέταρτο", λήξη "παρα τέταρτο" της επόμενης ώρας).

#### ΣΗΜΕΙΩΣΕΙΣ ΜΑΘΗΜΑΤΩΝ (ΘΕΩΡΙΑ):

[https://clab.math.auth.gr/sites/default/files/CS\\_Timeline.pdf](https://clab.math.auth.gr/sites/default/files/CS_Timeline.pdf)

1η διάλεξη (L1) :  
Διαφάνειες: <http://clab.math.auth.gr/sites/default/files/L1s22.pdf>

2η διάλεξη (L2) :  
Διαφάνειες:

3η διάλεξη (L3) :  
Διαφάνειες:

4η διάλεξη (L4) :  
Διαφάνειες:

5η και 6η διάλεξη (L5-6) :  
Διαφάνειες:

Tags: [Μαθηματικά](#) [Χρηστικά](#) [Εγχειρίδιο](#)

Εκτός από όπου αναφέρεται διαφορετικά, το περιεχόμενο του ιστοτόπου αδειοδοτείται με [Creative Commons Attribution \(Unported\) v3.0 License](#) [CC BY](#) [Open Data](#)

Copyright © 2022. Εργαστήριο Η/Υ Διατηρείται από Μονάδα Σηματολογικού Ιστού ΑΠΘ



# Εισαγωγή στον Προγραμματισμό

Εισαγωγική ενότητα (25%) :  
Βασικές έννοιες και τομείς  
της Επιστήμης Υπολογιστών  
1<sup>η</sup> ομιλία

Παναγιώτης Τζουνάκης

Άνοιξη 2022



# Περιεχόμενα

1. Ιστορικά στοιχεία και επισκόπηση  
αριθμητικά συστήματα (2/8/16-ικό), άλγεβρα Boole,  
αρχιτεκτονική Η/Υ, hard/firm/soft-ware, επίπεδα γλωσσών  
(μηχανή/assembly), μεταγλώτιση/διερμηνεία
2. Ανάλυση προβλημάτων, Αλγόριθμοι,  
Οργάνωση και δομές δεδομένων
3. Γλώσσες και μέθοδοι προγραμματισμού  
Στάδια, κύκλος ζωής λογισμικού



# Περιεχόμενα

## 4. Λειτουργικά Συστήματα, Πληροφοριακά Συστήματα

Εξοικείωση με DevC++

## 5. Τηλεπικοινωνίες και δίκτυα υπολογιστών

Δομή, τοπολογία, πρωτόκολλα, πρότυπα, οργάνωση διαχείριση και λειτουργία

## 6. Το Διαδίκτυο (Internet) και ο παγκόσμιος ιστός (World Wide Web)

Web1.0/2.0/3.0 , υπηρεσίες και εφαρμογές υπολογιστικού νέφους (cloud)





# Στόχος η ανάπτυξη:

1. αναλυτικής σκέψης και ικανότητα σύνθεσης
2. νοοτροπίας σαφήνειας και αυστηρής (τυπικής) διατύπωσης
3. δημιουργικότητας, σχεδιαστικής φαντασίας
4. μεθοδικότητας στην εργασία
5. δεξιοτήτων αλγοριθμικής σκέψης-προσέγγισης
6. ικανότητας (αποδοτικής) επίλυσης προβλημάτων με ευθύ (απλό) τρόπο και ελαχιστοποίηση πόρων





# Ακρογωνιαίοι λίθοι στην κατασκευή υπολογιστών

αναλυτικά βλ.

[https://en.wikipedia.org/wiki/History\\_of\\_computing\\_hardware](https://en.wikipedia.org/wiki/History_of_computing_hardware)

- 1642 Pascal +,-
- 1671 Leibniz +,-,\*,/
- 1827 Babbage: Polynomial  
Difference engine evaluation  
(method of finite  
differences)



- 1834 Babbage: Analytical Engine (Never completed) General purpose computation
- 1941 Zuse: Z3 General purpose computation
- 1944 Aiken: Harvard Mark I General purpose computation



- 1943-1946 ENIAC (Electronic Numerical Integrator and Calculator)  
University of Pennsylvania  
John Mauchly, Presper Eckert  
(30 tons, 18000 vacuum tubes!)
- Μηχανισμός των Αντικυθήρων (ανακαλύφτηκε το 1900)  
1971 Derek Price: Φορητός ημερολογιακός (ηλιακός-σεληνιακός) υπολογιστής  
IEEE Micro (1984): Ο πρώτος φορητός υπολογιστής!!



# Ιστορικά στοιχεία

## Ο μηχανισμός των Αντικυθήρων





# Ιστορικά στοιχεία

## Ο μηχανισμός των Αντικυθήρων

Κατασκευή 150 – 100 π.Χ.

Διαστάσεις 30x20x10 cm



© Rien van de Weijgaert



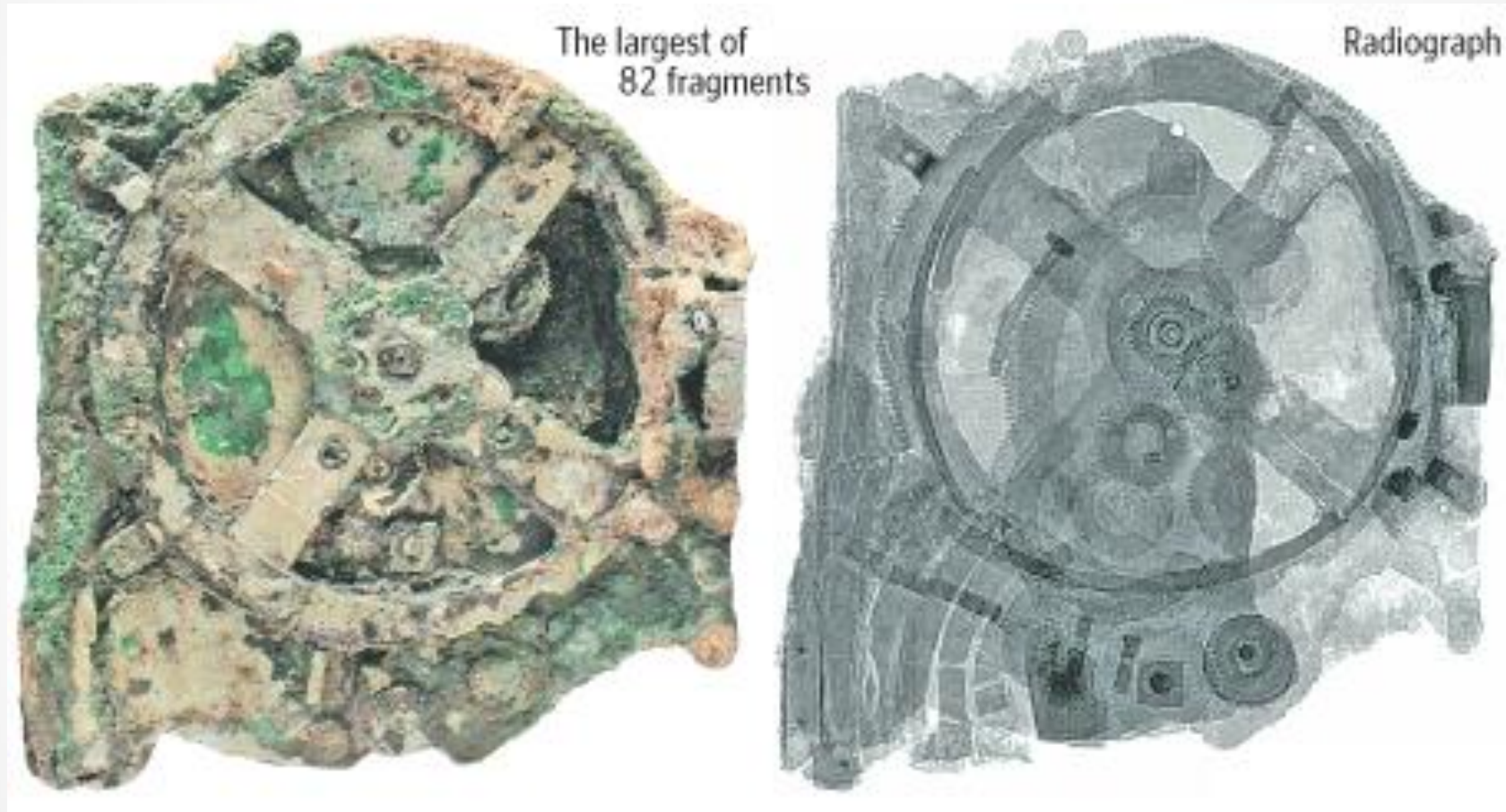
Τμήμα Μαθηματικών

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης



# Ιστορικά στοιχεία

## Ο μηχανισμός των Αντικυθήρων





# Ιστορικά στοιχεία

Ο μηχανισμός των

Αντικυθήρων



© Antikythera Mechanism Research Project



Τμήμα Μαθηματικών

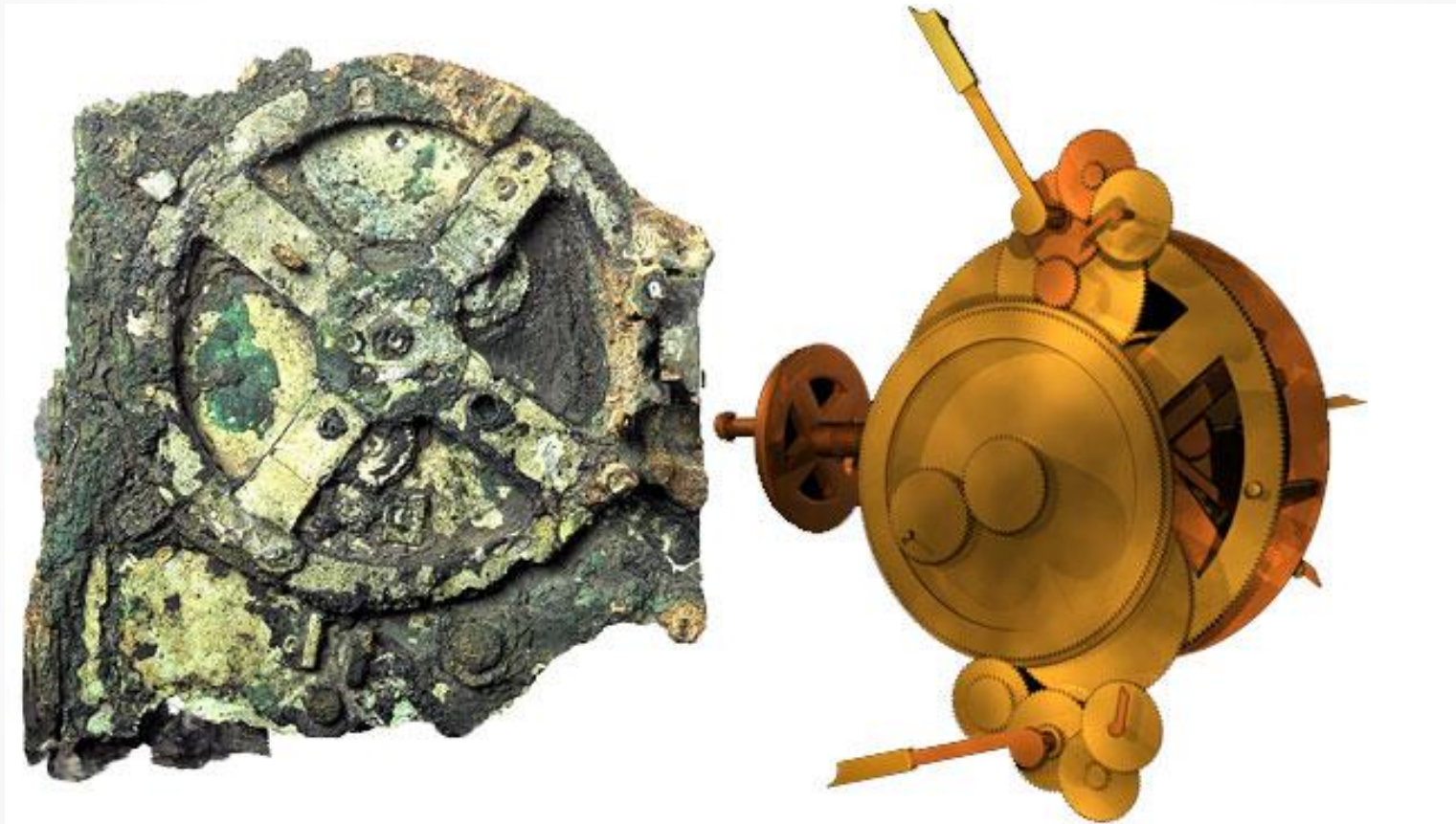
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης





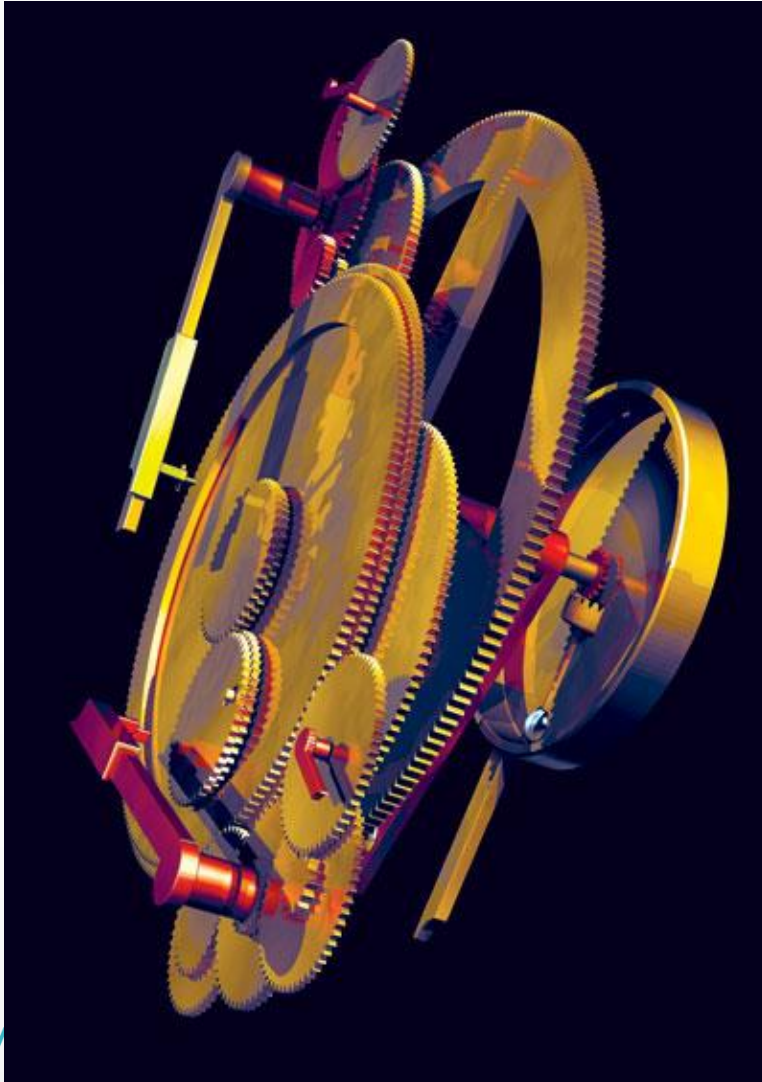
# Ιστορικά στοιχεία

## Ο μηχανισμός των Αντικυθήρων



# Ιστορικά στοιχεία

## Ο μηχανισμός των Αντικυθήρων



F. Charette

High tech from Ancient Greece

Nature, vol 444, 30 November 2006, 551-552

J. Seiradakis, et al.

Decoding the ancient Greek astronomical  
calculator known as the Antikythera

Mechanism

Nature, vol. 444, 30 November 2006, 587-591

T. Freeth, et al.

Calendars with Olympiad display and eclipse prediction on  
the Antikythera Mechanism

Nature, vol. 454, 31 July 2008, 614-617

<http://www.antikythera-mechanism.gr>

<http://www.antikythera-mechanism.gr/media/movies>



# Ιστορικά στοιχεία

<https://ethw.org/w/images/1/19/Timeline.pdf>



[https://en.wikipedia.org/wiki/History\\_of\\_computer\\_science](https://en.wikipedia.org/wiki/History_of_computer_science)

[https://en.wikipedia.org/wiki/Timeline\\_of\\_computing](https://en.wikipedia.org/wiki/Timeline_of_computing)

[https://en.wikipedia.org/wiki/List\\_of\\_pioneers\\_in\\_computer\\_science](https://en.wikipedia.org/wiki/List_of_pioneers_in_computer_science)



Τμήμα Μαθηματικών

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης



# Ιστορικά στοιχεία

Η Επιστήμη Υπολογιστών σήμερα:

- Θεωρητική
  - Ανάλυση Αλγορίθμων
  - Θεωρία Υπολογισιμότητας (Computability)
  - Θεωρία Πολυπλοκότητας (Complexity)
  - Θεωρία Γλωσσών
  - Formal Methods (Specification & Verification)
  - Θεωρία Πληροφορίας, **κρυπτογραφία**
  - ...



# Ιστορικά στοιχεία

Η Επιστήμη Υπολογιστών σήμερα:

- Εφαρμοσμένη

- Υλικό (hardware) : ψηφιακή σχεδίαση, αρχιτεκτονική Η/Υ
- Λογισμικό : Εφαρμογές, Λειτουργικά Συστήματα, ...
- Τεχνητή Νοημοσύνη
- Δίκτυα Δεδομένων
- Πληροφοριακά Συστήματα, Βάσεις δεδομένων/γνώσης
- Γραφικά, διεπαφή χρήστη,
- Ασφάλεια, **κρυπτογραφία**
- ...

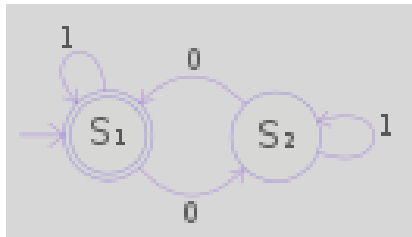




# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

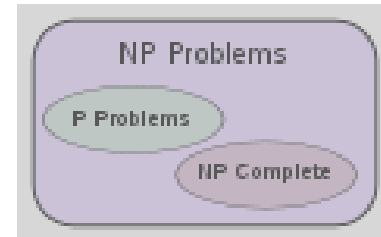
## Theory of computation



Automata theory

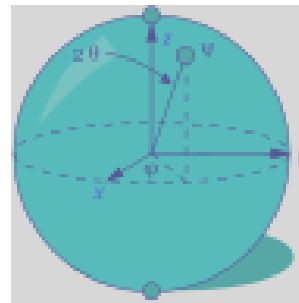
$$M = \{X : X \notin X\}$$

Computability theory



Computational complexity theory

GNITIRW-TERCES



Quantum computing theory



Digital circuit theory

Cryptography

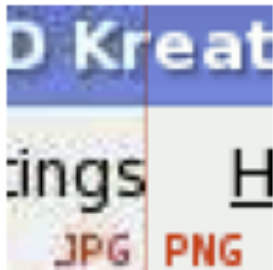




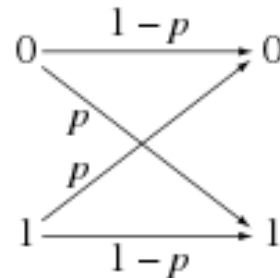
# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

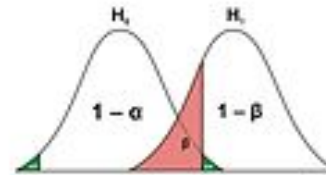
Information and coding theory



Source coding



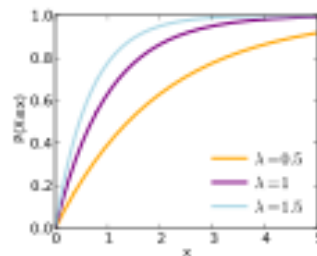
Channel capacity



Signal detection theory

```
gcc -O3 foo.c
```

Algorithmic information theory



Information geometry

$$\hat{\theta} = \operatorname{argmax}_{\theta} P(x|\theta)$$

Estimation theory

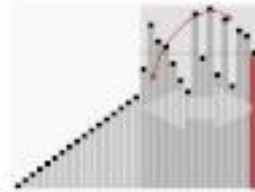


# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

Data structures and algorithms

$O(n^2)$



12 → 99 → 37 → ☒



Analysis of algorithms

Algorithms

Data structures

Combinatorial optimization

Computational geometry



Τμήμα Μαθηματικών

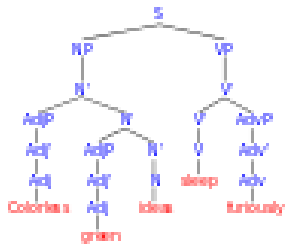
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης



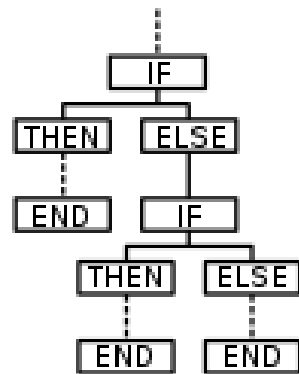
# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

## Programming language theory



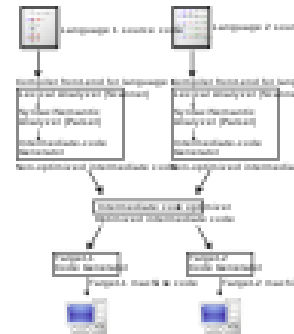
Formal languages



Formal semantics

$\Gamma \vdash x : \text{Int}$

Type theory



Compiler design

```
def add(x, y):  
    return x + y  
  
def multiply(x, y):  
    return x * y  
  
def main():  
    x = 10  
    y = 5  
    z = add(x, y)  
    print(z)  
    w = multiply(x, y)  
    print(w)  
  
if __name__ == '__main__':  
    main()
```

Programming languages

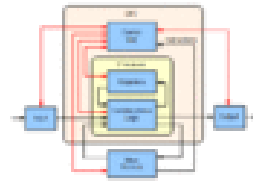




# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

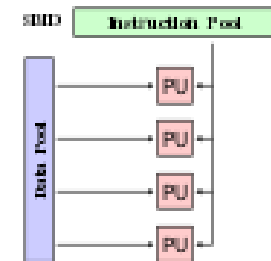
Computer  
architecture  
& engineering



Processor



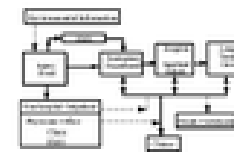
Microarchitecture



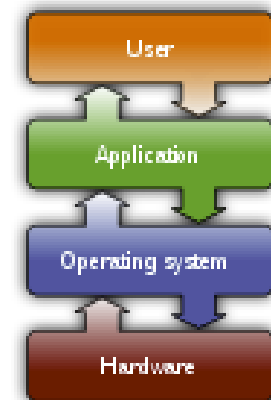
Multiprocessing



Ubiquitous computing



Systems architecture



Operating systems



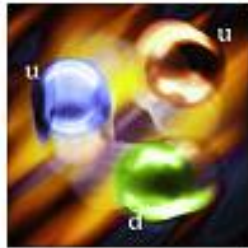
# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

Scientific computing and simulation



Numerical analysis



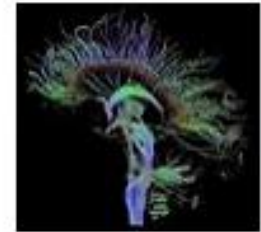
Computational physics



Computational chemistry



Bioinformatics



Computational neuroscience



# Η Επιστήμη Υπολογιστών με εικόνες

βλ. [https://en.wikipedia.org/wiki/Computer\\_science](https://en.wikipedia.org/wiki/Computer_science)

Artificial intelligence



Machine learning



Computer vision

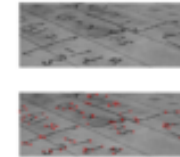
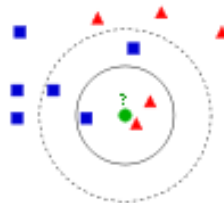
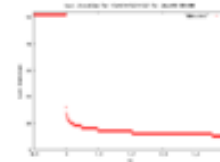


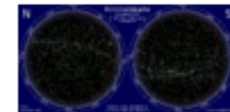
Image processing



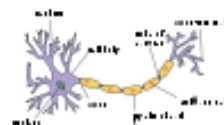
Pattern recognition



Data mining



Evolutionary computation



Knowledge representation and reasoning

abcdefghijklmnopqrstuvwxyz  
ABCDEFGHIJKLMNOPQRSTUVWXYZ  
OPQRSTUVWXYZ

Natural language processing



Robotics





# Η Επιστήμη Υπολογιστών σήμερα:

Οι υπολογιστές είναι ο καταλύτης για την 4<sup>η</sup> Βιομηχανική Επανάσταση (4IR)

[https://en.wikipedia.org/wiki/Fourth\\_Industrial\\_Revolution](https://en.wikipedia.org/wiki/Fourth_Industrial_Revolution)

<https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/>

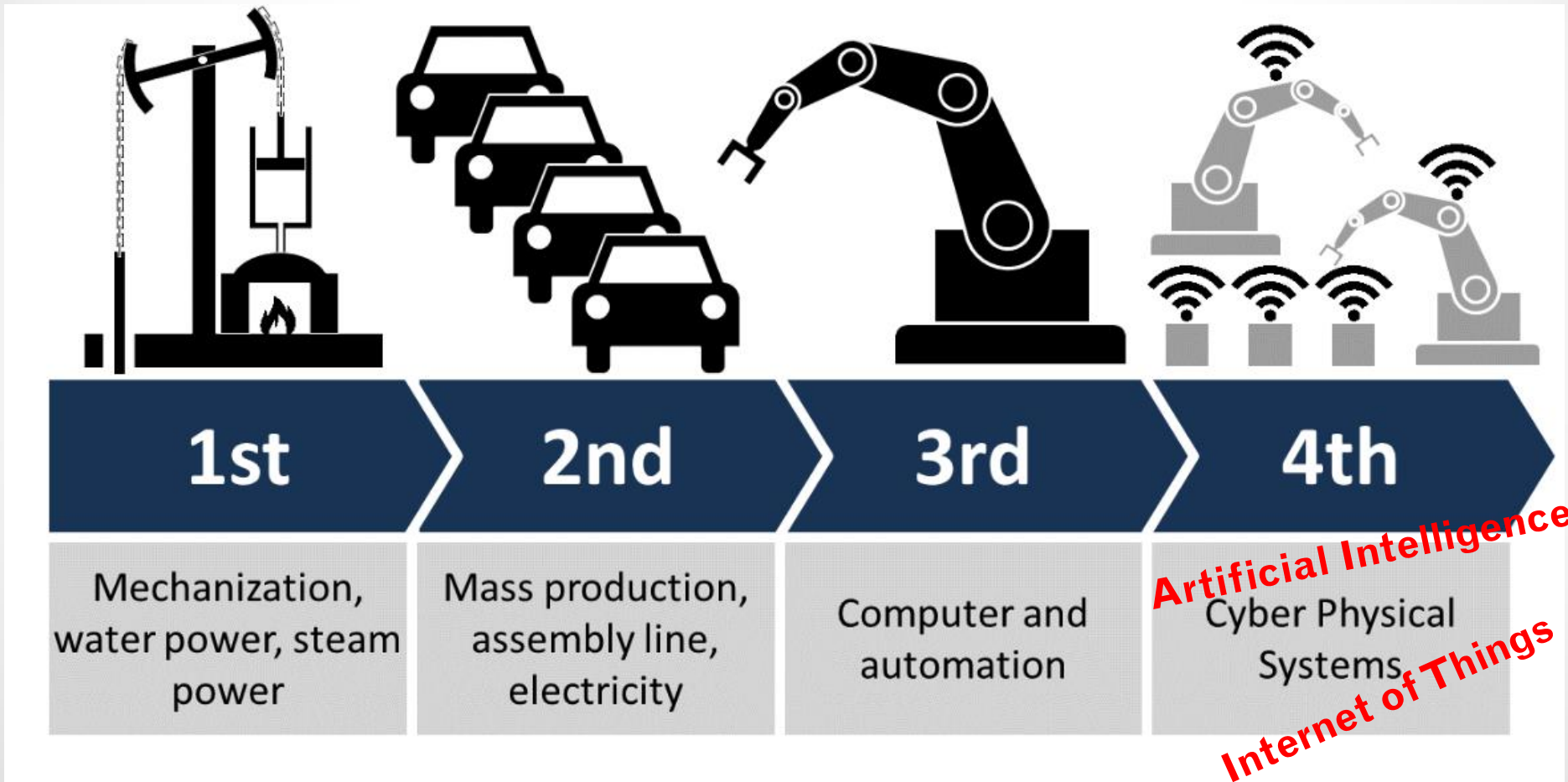


Image Credit: "Christoph Roser at [AllAboutLean.com](http://AllAboutLean.com)"

Τμήμα Μαθηματικών

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης



# Αριθμητικά συστήματα

**Δεκαδικό** σύστημα (βάση είναι το 10, δεκαδικά ψηφία) :

$$(d_n \dots d_2 d_1 d_0)_{10} \stackrel{\text{def}}{=} d_n \times 10^n + \dots + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0$$

όπου  $d_0, d_1, d_2, \dots, d_n \in \{0, 1, \dots, 9\}$

$$(532)_{10} \stackrel{\text{def}}{=} 5 \times 100 + 3 \times 10 + 2 \times 1 \stackrel{\text{def}}{=} 5 \times 10^2 + 3 \times 10^1 + 2 \times 10^0$$

**Δυαδικό** σύστημα (βάση είναι το 2, δυαδικά ψηφία – binary digits (bits)) :

$$(d_n \dots d_2 d_1 d_0)_2 \stackrel{\text{def}}{=} d_n \times 2^n + \dots + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0$$

όπου  $d_0, d_1, d_2, \dots, d_n \in \{0, 1\}$

$$(532)_{10} \stackrel{\text{def}}{=} (1000010100)_2 \stackrel{\text{def}}{=} 1 \times 512 + \dots + 1 \times 16 + 0 \times 8 + 1 \times 4 + 0 \times 2 + 0 \times 1$$
$$= 1 \times 2^9 + 1 \times 2^4 + 1 \times 2^2$$



# Αριθμητικά συστήματα

**Οκταδικό** σύστημα (βάση είναι το 8, οκταδικά ψηφία - oct) :

$$(d_n \dots d_2 d_1 d_0)_8 \stackrel{\text{def}}{=} d_n \times 8^n + \dots + d_2 \times 8^2 + d_1 \times 8^1 + d_0 \times 8^0$$

όπου  $d_0, d_1, d_2, \dots, d_n \in \{0, 1, 2, 3, 4, 5, 6, 7\}$

$$(532)_8 \stackrel{\text{def}}{=} 5 \times 64 + 3 \times 8 + 2 \times 1 \stackrel{\text{def}}{=} 5 \times 8^2 + 3 \times 8^1 + 2 \times 8^0$$

**ΠΡΟΣΟΧΗ!!!** :  $(532)_8 \neq (532)_{10}$

binary 2 octal : 3 bits στη σειρά αντικαθίστανται από το αντίστοιχο οκταδικό ψηφίο, αρχίζοντας από το λιγότερο σημαντικό bit. Συμπληρώνουμε μηδενικά αριστερά του περισσότερο σημαντικού bit εφόσον χρειάζεται.

$$(1101001010)_2 = (1512)_8$$



# Αριθμητικά συστήματα

**Δεκαεξαδικό** σύστημα (βάση είναι το 16, δεκαεξαδικά ψηφία - hex) :

$$(d_n \dots d_2 d_1 d_0)_2 \stackrel{\text{def}}{=} d_n \times 16^n + \dots + d_2 \times 16^2 + d_1 \times 16^1 + d_0 \times 16^0$$

όπου  $d_0, d_1, d_2, \dots, d_n \in \{0, 1, \dots, 9, A, B, C, D, E, F\}$

$$(532)_{16} \stackrel{\text{def}}{=} 5 \times 256 + 3 \times 16 + 2 \times 1 \stackrel{\text{def}}{=} 5 \times 16^2 + 3 \times 16^1 + 2 \times 16^0$$

binary 2 hexadecimal : 4 bits στη σειρά αντικαθίστανται από το αντίστοιχο δεκαεξαδικό ψηφίο, αρχίζοντας από το λιγότερο σημαντικό bit. Συμπληρώνουμε μηδενικά αριστερά του περισσότερο σημαντικού bit εφόσον χρειάζεται.

$$(1101001010)_2 = (34A)_{16} = (1512)_8$$



# Αριθμητικά συστήματα

Αριθμητικές πράξεις : (business) as usual!

Πρόσθεση:

$$\begin{array}{r} 1\ 1\ 1\ 1\ 1 \quad (\text{carried digits}) \\ 0\ 1\ 1\ 0\ 1 \\ +\ 1\ 0\ 1\ 1\ 1 \\ \hline = 1\ 0\ 0\ 1\ 0\ 0 = 36 \end{array}$$

Πολλαπλασιασμός:

$$\begin{array}{r} 1\ 0\ 1\ 1 \quad (A) \\ \times 1\ 0\ 1\ 0 \quad (B) \\ \hline 0\ 0\ 0\ 0 \quad \leftarrow \text{Corresponds to the rightmost 'zero' in } B \\ + 1\ 0\ 1\ 1 \quad \leftarrow \text{Corresponds to the next 'one' in } B \\ + 0\ 0\ 0\ 0 \\ + 1\ 0\ 1\ 1 \\ \hline = 1\ 1\ 0\ 1\ 1\ 1\ 0 \end{array}$$



# Αριθμητικά συστήματα

Δυαδικοί αριθμοί με πρόσημο : Το συμπλήρωμα ως προς 2

Ορισμός: Για δυαδικό αριθμό  $x$  που αναπαριστάται με  $N$  το πλήθος bits, είναι ο δυαδικός αριθμός  $y$ , τέτοιος ώστε  $x + y = 2^N$

Υπολογισμός: Αντιστρέφουμε όλα τα bits (σημ: αυτό ονομάζεται «συμπλήρωμα ως προς 1») και κατόπιν προσθέτουμε 1.

**Three-bit signed integers**

Decimal value	Two's-complement Representation
0	000
1	001
2	010
3	011
-4	100
-3	101
-2	110
-1	111





# Αριθμητικά συστήματα

[https://en.wikipedia.org/wiki/Two's\\_complement](https://en.wikipedia.org/wiki/Two's_complement)

[https://en.wikipedia.org/wiki/Ones'\\_complement](https://en.wikipedia.org/wiki/Ones'_complement)

Eight-bit signed integers

Decimal value ↕	Two's-complement Representation ↕
0	0000 0000
1	0000 0001
2	0000 0010
126	0111 1110
127	0111 1111
-128	1000 0000
-127	1000 0001
-126	1000 0010
-2	1111 1110
-1	1111 1111



# Άλγεβρα Boole

[https://en.wikipedia.org/wiki/Boolean\\_algebra](https://en.wikipedia.org/wiki/Boolean_algebra)

Βασικές πράξεις:

ΚΑΙ – σύζευξη (AND – conjunction) :  $x \wedge y$  ( $x$  AND  $y$ )

Η – διάζευξη (OR – disjunction) :  $x \vee y$  ( $x$  OR  $y$ )

ΟΧΙ - άρνηση (NOT – negation) :  $\neg x$  (NOT  $x$ ,  $\bar{x}$  ή  $!x$ )

Πίνακες αληθείας

$x$	$y$	$x \wedge y$	$x \vee y$	$x$	$\neg x$
0	0	0	0	0	1
1	0	0	1	1	0
0	1	0	1		
1	1	1	1		



# Άλγεβρα Boole

[https://en.wikipedia.org/wiki/Boolean\\_algebra](https://en.wikipedia.org/wiki/Boolean_algebra)

δευτερεύουσες πράξεις: Υλική συνεπαγωγή, Αποκλειστικό Η (Exclusive OR – XOR), Ισοδυναμία (equivalence)

$$x \rightarrow y = \neg x \vee y$$

$$x \oplus y = \neg(x \equiv y) = (x \vee y) \wedge (\neg x \vee \neg y) = (x \wedge \neg y) \vee (\neg x \wedge y)$$

$$x \equiv y = \neg(x \oplus y) = (x \wedge y) \vee (\neg x \wedge \neg y)$$

Πίνακες αληθείας

$x$	$y$	$x \rightarrow y$	$x \oplus y$	$x \equiv y$
0	0	1	0	1
1	0	0	1	0
0	1	1	1	0
1	1	1	0	1



# Άλγεβρα Boole

Νόμοι (κατ' αναλογία με συνήθη άλγεβρα με + αντί  $\vee$  και \* αντί  $\wedge$ )

Associativity of  $\vee$ :

$$x \vee (y \vee z) = (x \vee y) \vee z$$

Associativity of  $\wedge$ :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

Commutativity of  $\vee$ :

$$x \vee y = y \vee x$$

Commutativity of  $\wedge$ :

$$x \wedge y = y \wedge x$$

Distributivity of  $\wedge$  over  $\vee$ :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

Identity for  $\vee$ :

$$x \vee 0 = x$$

Identity for  $\wedge$ :

$$x \wedge 1 = x$$

Annihilator for  $\wedge$ :

$$x \wedge 0 = 0$$



# Άλγεβρα Boole

Νόμοι (δεν ισχύουν στη συνήθη άλγεβρα με + , \*)

Annihilator for  $\vee$ :  $x \vee 1 = 1$

Idempotence of  $\vee$ :  $x \vee x = x$

Idempotence of  $\wedge$ :  $x \wedge x = x$

Absorption 1:  $x \wedge (x \vee y) = x$

Absorption 2:  $x \vee (x \wedge y) = x$

Distributivity of  $\vee$  over  $\wedge$ :  $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$



# Άλγεβρα Boole

Κι άλλοι νόμοι!

Complementation 1

$$x \wedge \neg x = 0$$

Για το συμπλήρωμα :

Complementation 2

$$x \vee \neg x = 1$$

Διπλή άρνηση:

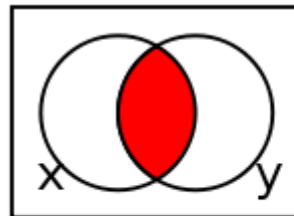
$$\neg(\neg x) = x$$

Νόμοι De Morgan:

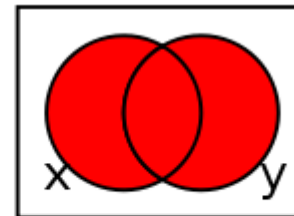
$$\neg x \wedge \neg y = \neg(x \vee y)$$

$$\neg x \vee \neg y = \neg(x \wedge y)$$

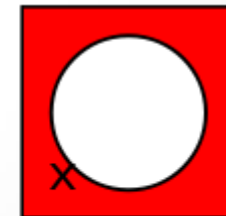
Διαγράμματα Venn για πράξεις στην άλγεβρα Boole :



$$x \wedge y$$



$$x \vee y$$



$$\neg x$$





# Άλγεβρα Boole και ψηφιακές λογικές πύλες

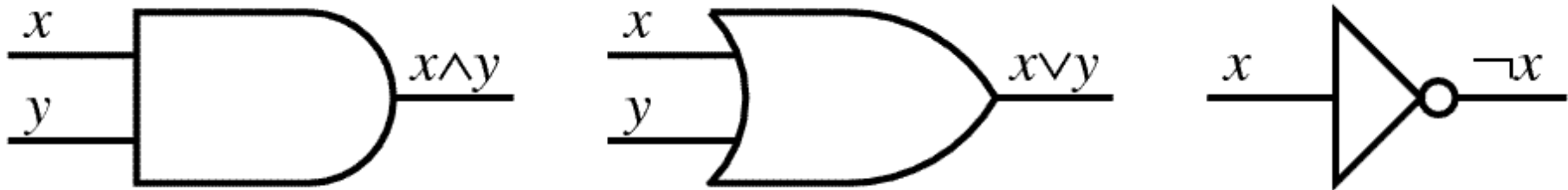


Figure 3. Logic gates

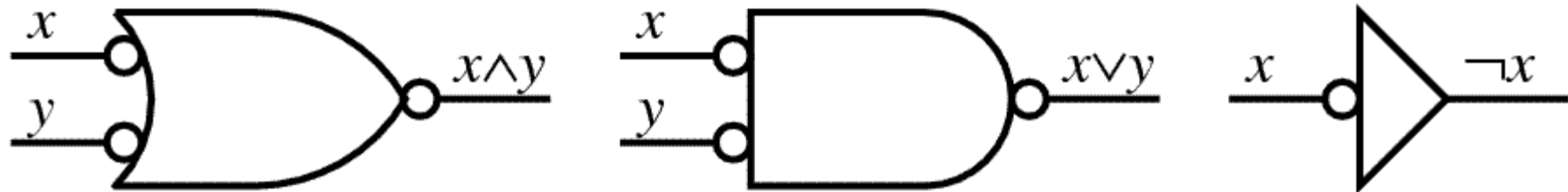


Figure 4. De Morgan equivalents

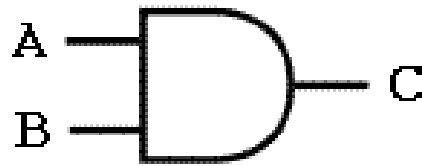


# Αντιστρεψιμότητα (reversibility) υπολογισμών

- Φυσική και λογική αντιστρεψιμότητα

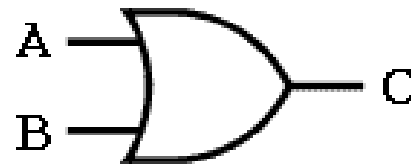
[https://en.wikipedia.org/wiki/Reversible\\_computing](https://en.wikipedia.org/wiki/Reversible_computing)

AND



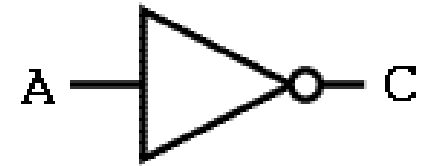
Inputs		Output
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1 ✓

OR



Inputs		Output
A	B	C
0	0	0 ✓
0	1	1
1	0	1
1	1	1

NOT



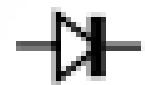
Input	Output
A	C
0	1 ✓
1	0 ✓



Δείτε την αρχή και το όριο του Landauer [https://en.wikipedia.org/wiki/Landauer's\\_principle](https://en.wikipedia.org/wiki/Landauer's_principle)



# Ψηφιακά κυκλώματα



Diode



Capacitor



Inductor



Resistor



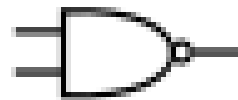
DC voltage source



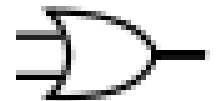
AC voltage source



And gate



Nand gate



Or gate



Nor gate



Xor gate

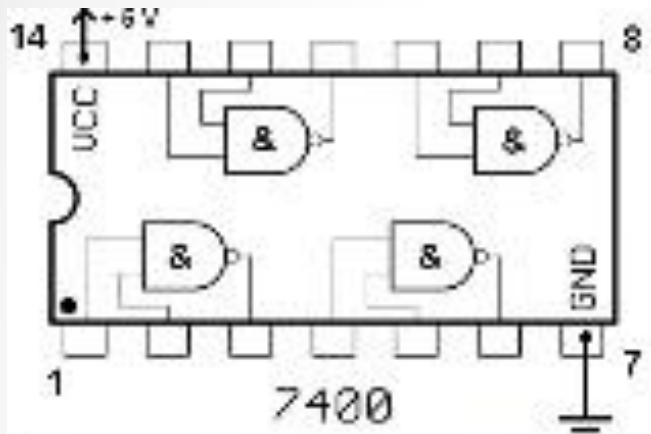


Inverter  
(Not gate)

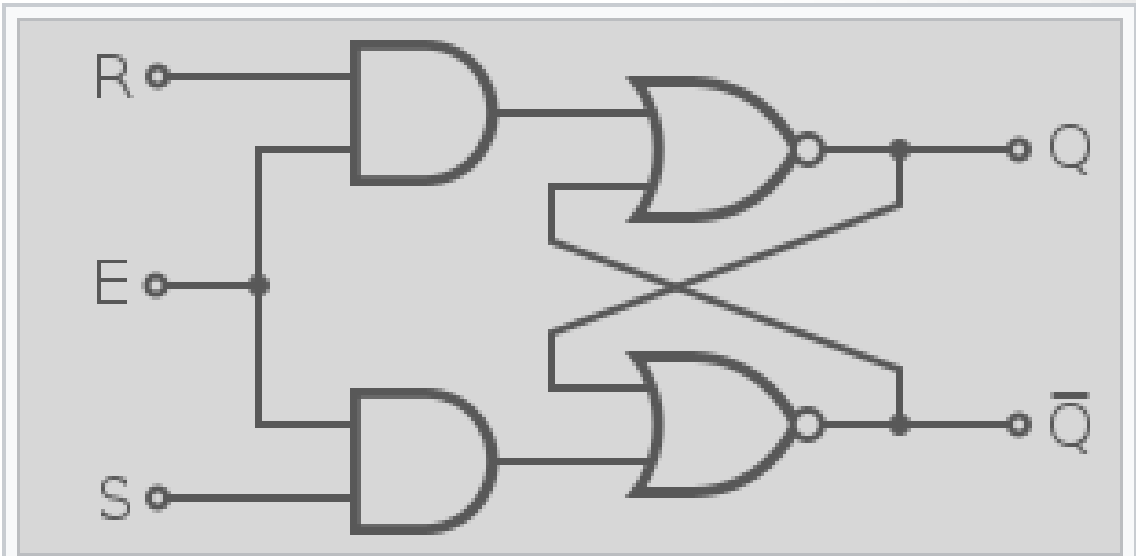
Συνήθη σύμβολα σε σχηματικά διαγράμματα κυκλωμάτων



# Ψηφιακά κυκλώματα



Ακολουθιακό κύκλωμα



A gated SR latch circuit diagram constructed from AND gates (on left) and NOR gates (on right).

«τσιπάκι» (αριστερά) και μανταλωτής (δεξιά)

βλ. [https://en.wikipedia.org/wiki/Logic\\_gate#Symbols](https://en.wikipedia.org/wiki/Logic_gate#Symbols) και

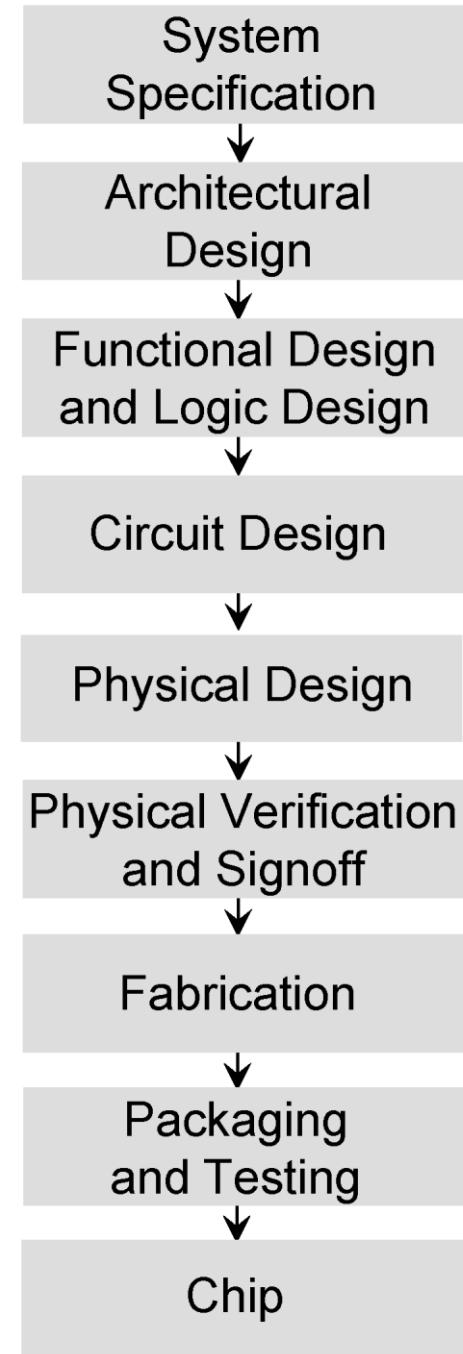
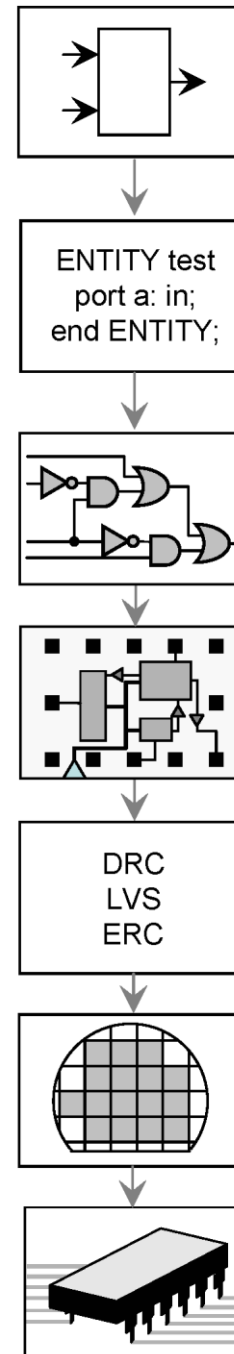
[https://en.wikipedia.org/wiki/Flip-flop\\_\(electronics\)](https://en.wikipedia.org/wiki/Flip-flop_(electronics)) αντίστοιχα



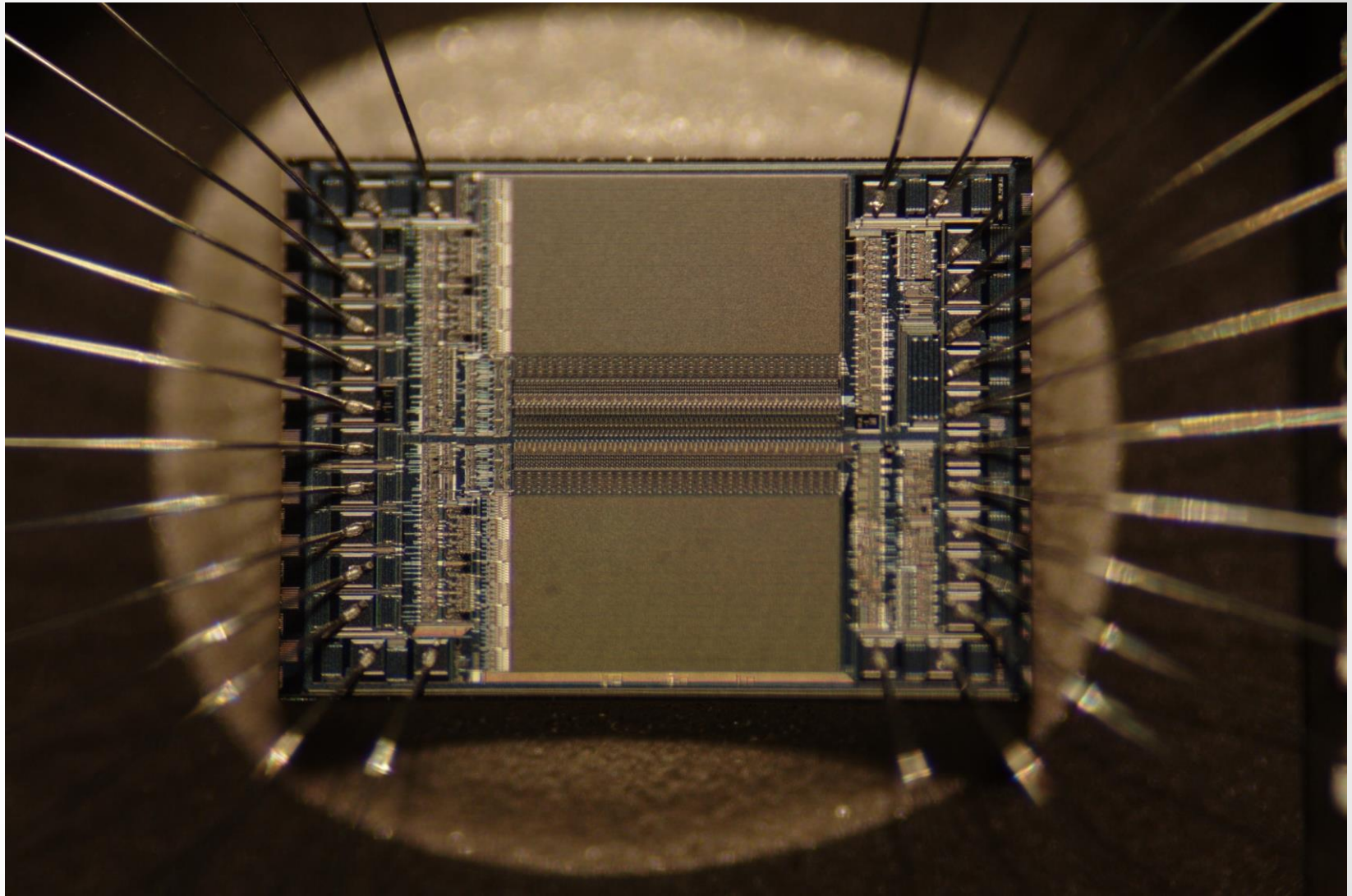
# Ψηφιακή Σχεδίαση

Στάδια κατασκευής  
ολοκληρωμένου κυκλώματος  
(chip)

[https://en.wikipedia.org/wiki/Integrated\\_circuit\\_design](https://en.wikipedia.org/wiki/Integrated_circuit_design)



# Ψηφιακή σχεδίαση



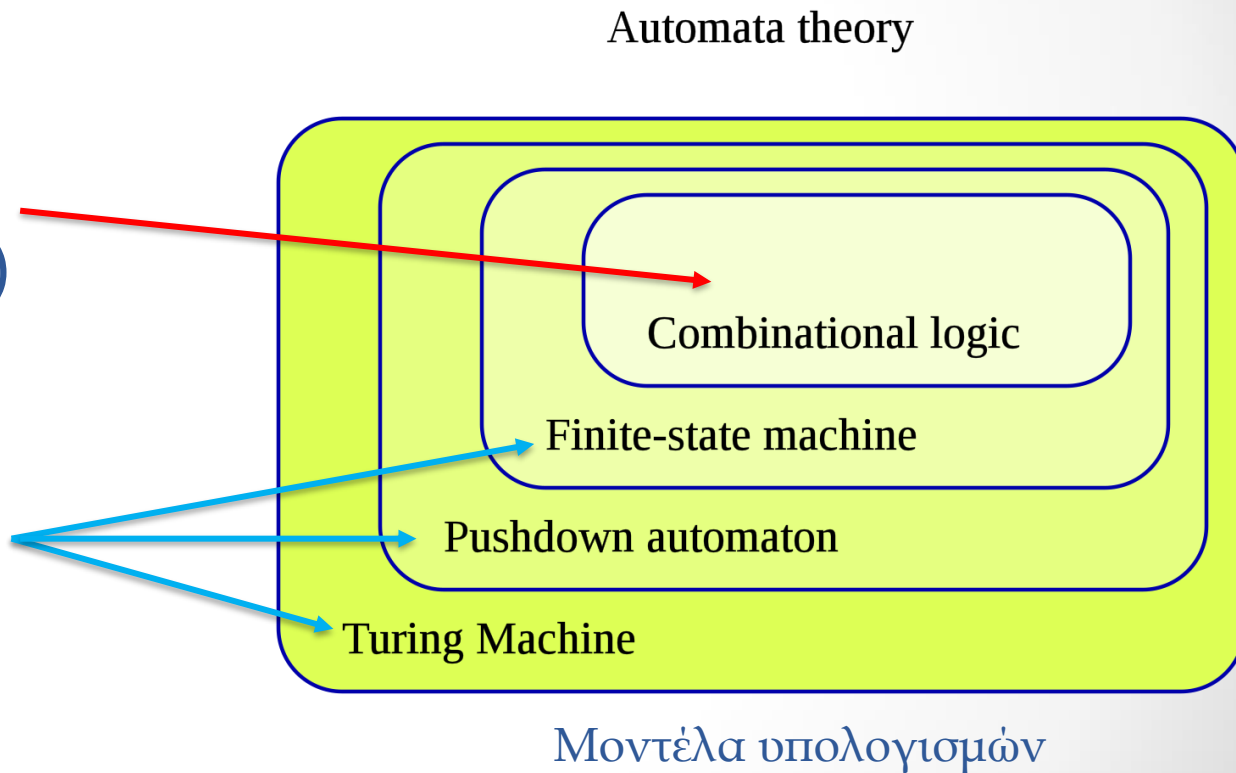
Integrated circuit from an EPROM memory microchip showing the memory blocks, the supporting circuitry and the fine silver wires which connect the integrated circuit die to the legs of the packaging



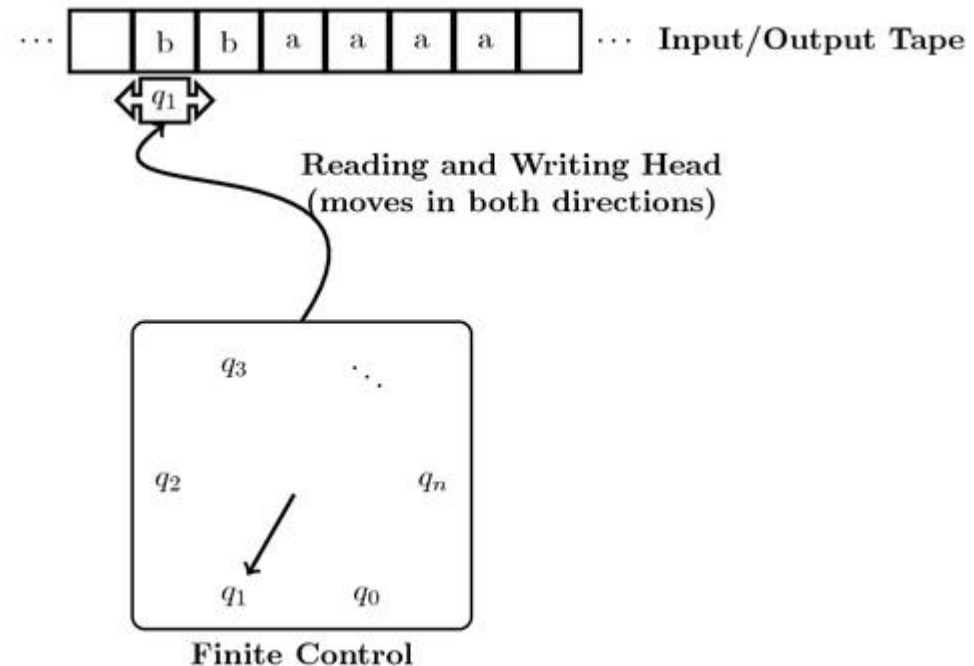
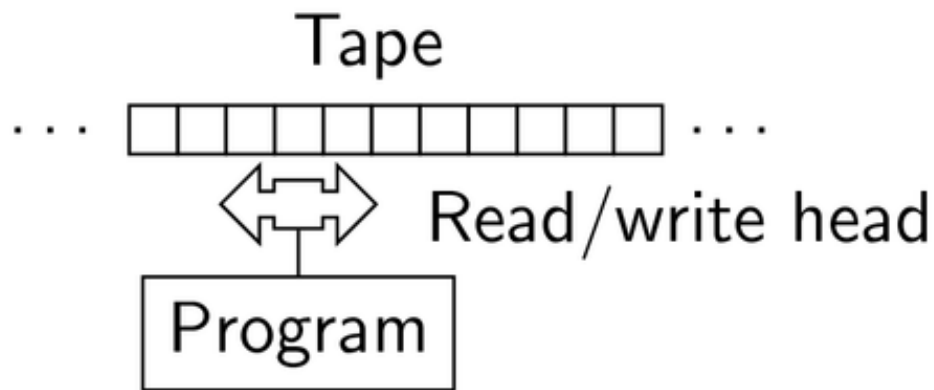


# Από την ψηφιακή σχεδίαση στην αρχιτεκτονική Η/Υ

- Συνδυαστικά κυκλώματα (συνδυαστική λογική – combinational logic)
- Ακολουθιακά κυκλώματα (ακολουθιακή λογική – sequential logic) υλοποιεί μνήμη!



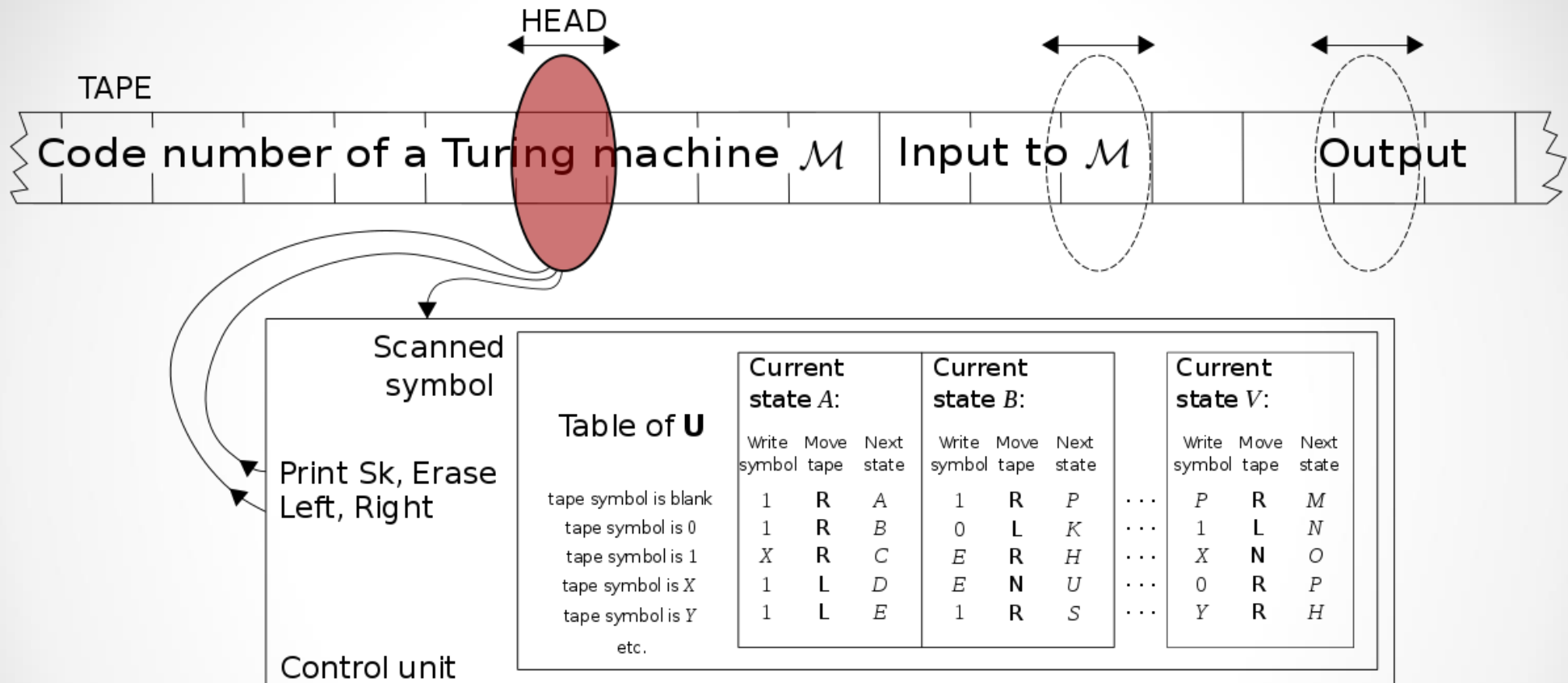
# Η μηχανή του Turing



[https://en.wikipedia.org/wiki/Turing\\_machine](https://en.wikipedia.org/wiki/Turing_machine)



# Universal Turing Machine

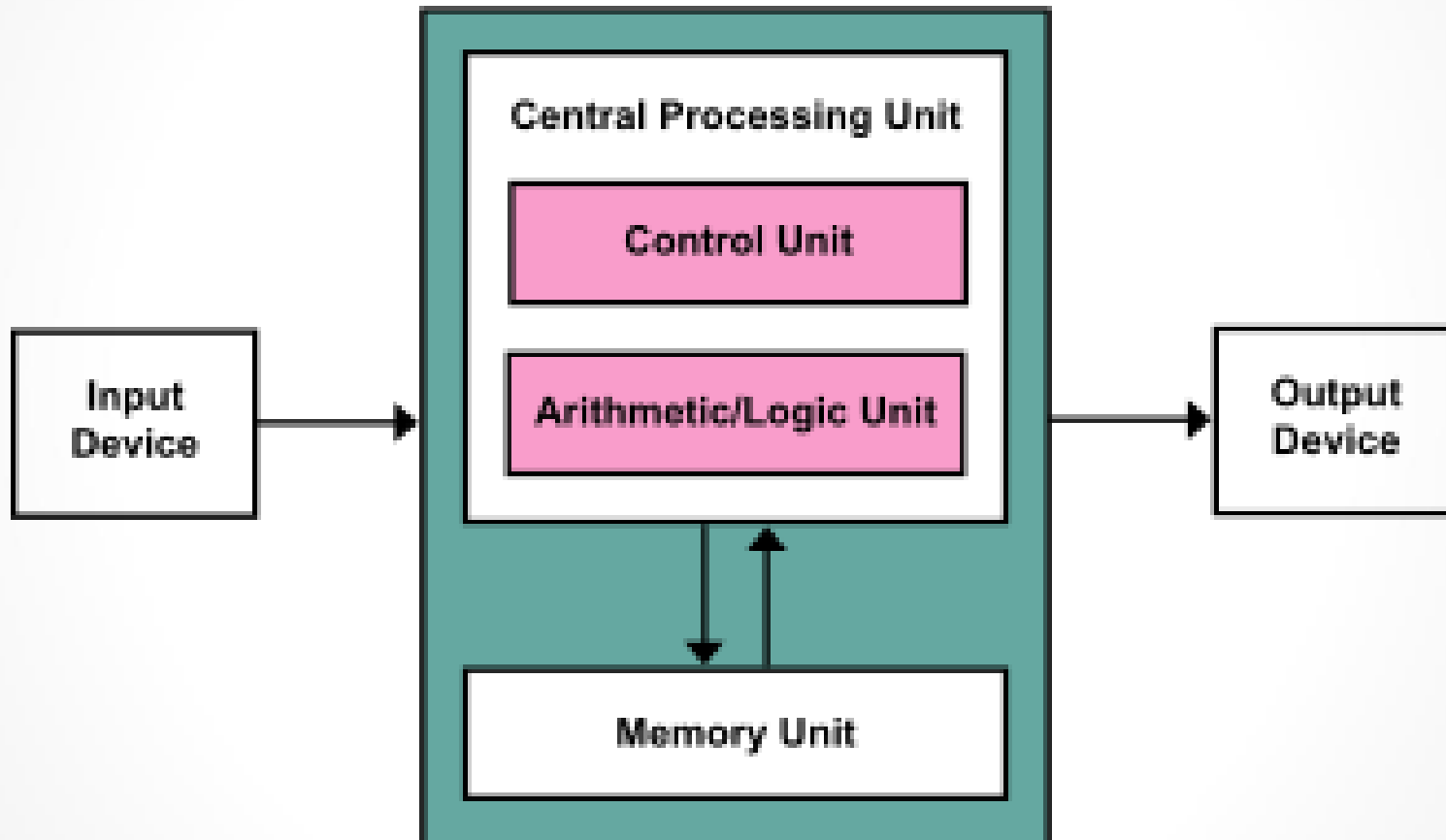


[https://en.wikipedia.org/wiki/Universal\\_Turing\\_machine](https://en.wikipedia.org/wiki/Universal_Turing_machine)

"...the origin of the idea of a stored-program computer used by John von Neumann in 1946 for the "Electronic Computing Instrument" that now bears von Neumann's name: the von Neumann architecture.<sup>[1]</sup>"



# Αρχιτεκτονική Η/Υ



Αρχιτεκτονική **von Neumann**



ΟΔΗΓΟΣ ΣΠΟΥΔΩΝ  
2021 - 2022



## Αρχιτεκτονική von Neumann

Οδηγός σπουδών 2021-22

<https://drive.google.com/file/d/1MDznpDkSSWurOSABAUtrrUB8csyNgvEh/view>



# Αρχιτεκτονική Η/Υ

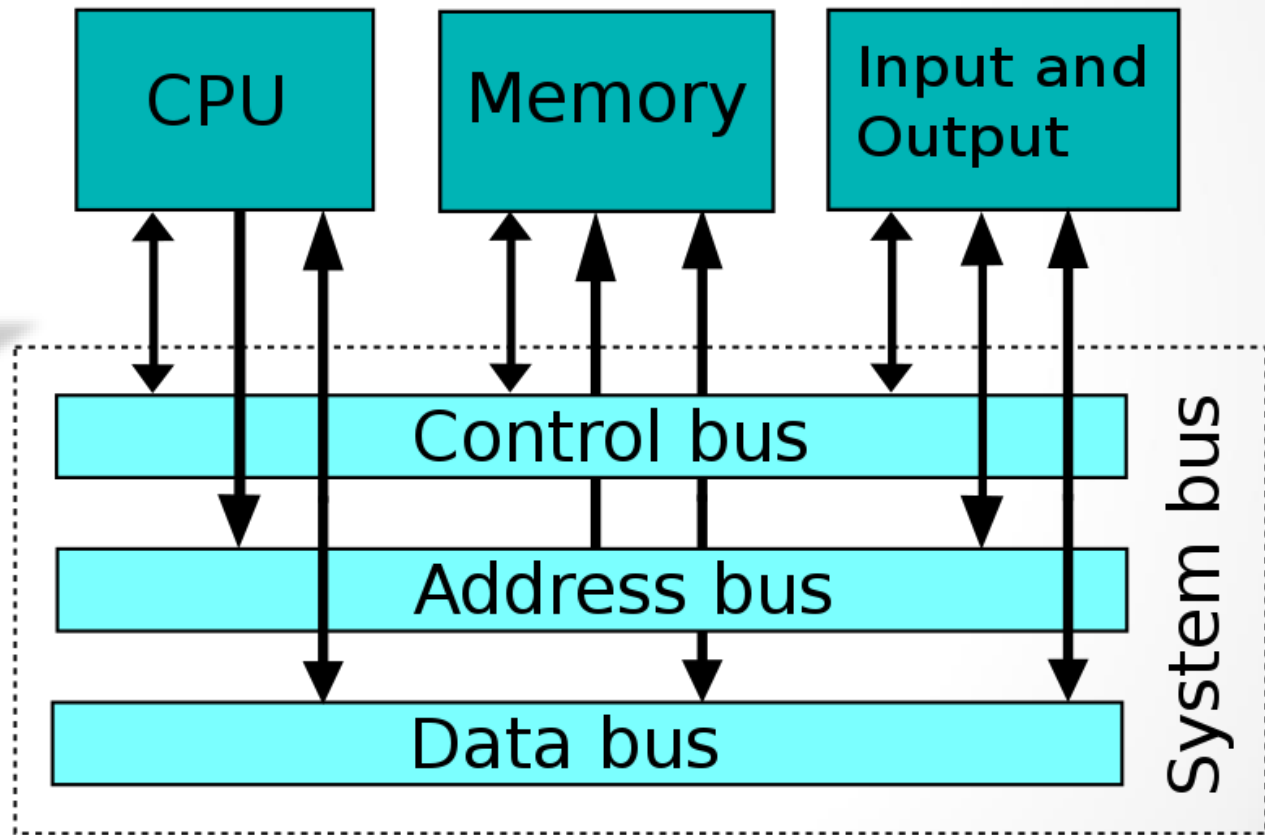
προσκόμιση (fetch)

αποκωδικοποίηση  
(decode)

εκτέλεση (execution)

αποθήκευση (store)

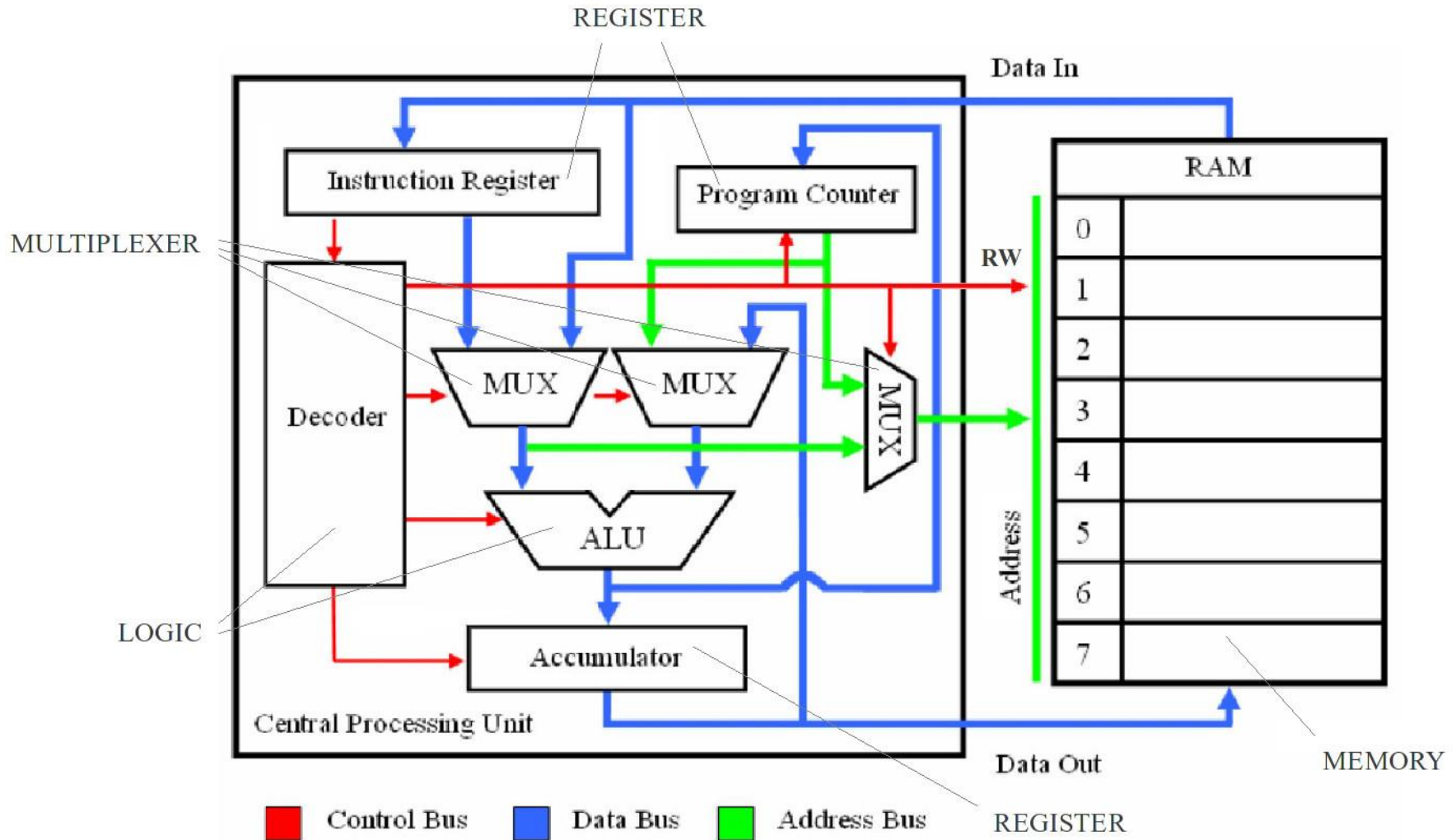
Απλουστευμένη  
παρουσίαση  
σταδίων εκτέλεσης  
εντολής μηχανής  
στη CPU



Αρχιτεκτονική von Neumann



# Αρχιτεκτονική Η/Υ

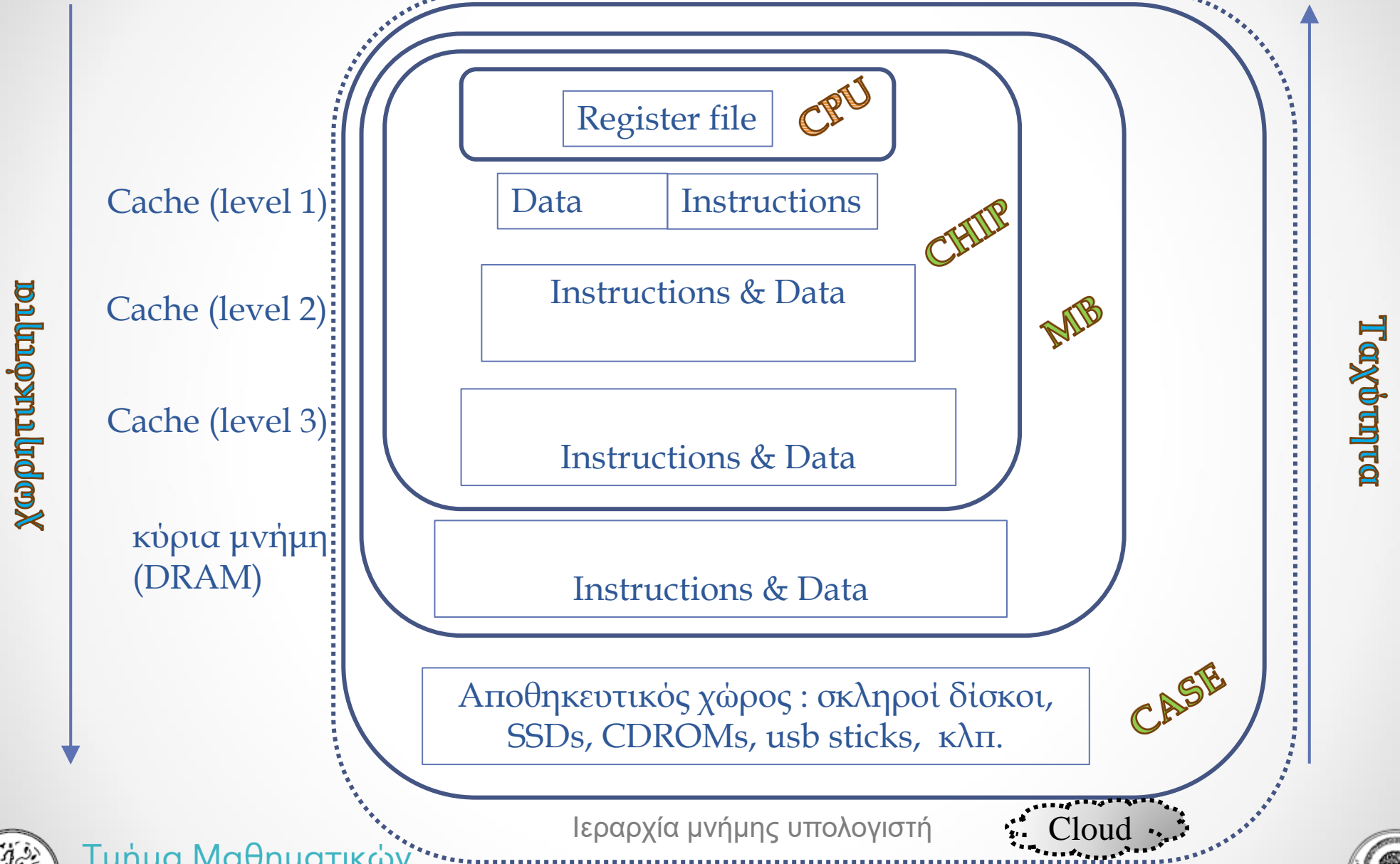


Πηγή : [http://www.simplecpudesign.com/simple\\_cpu\\_v1/index.html](http://www.simplecpudesign.com/simple_cpu_v1/index.html)



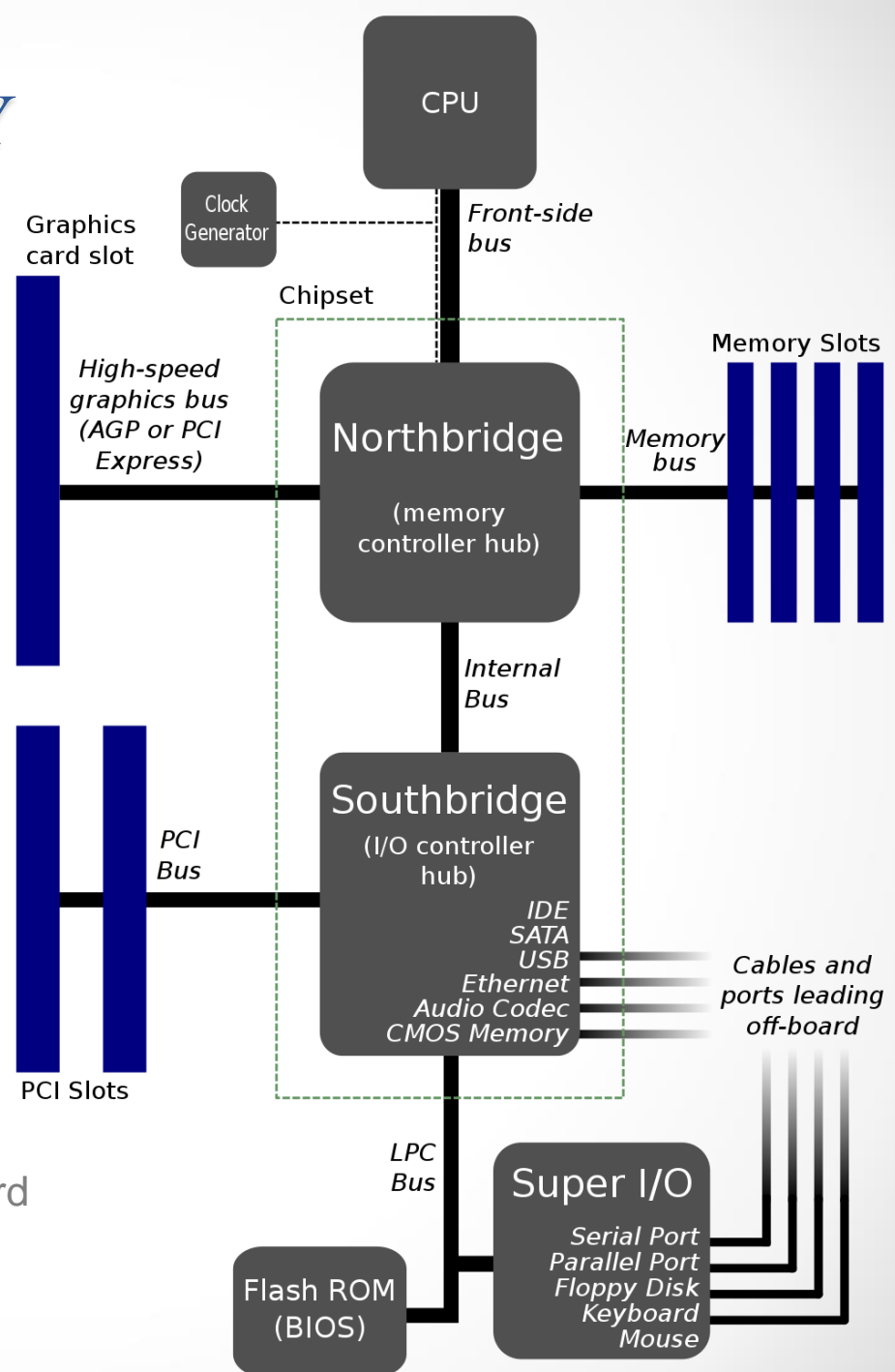


# Αρχιτεκτονική Η/Υ



# Αρχιτεκτονική Η/Υ

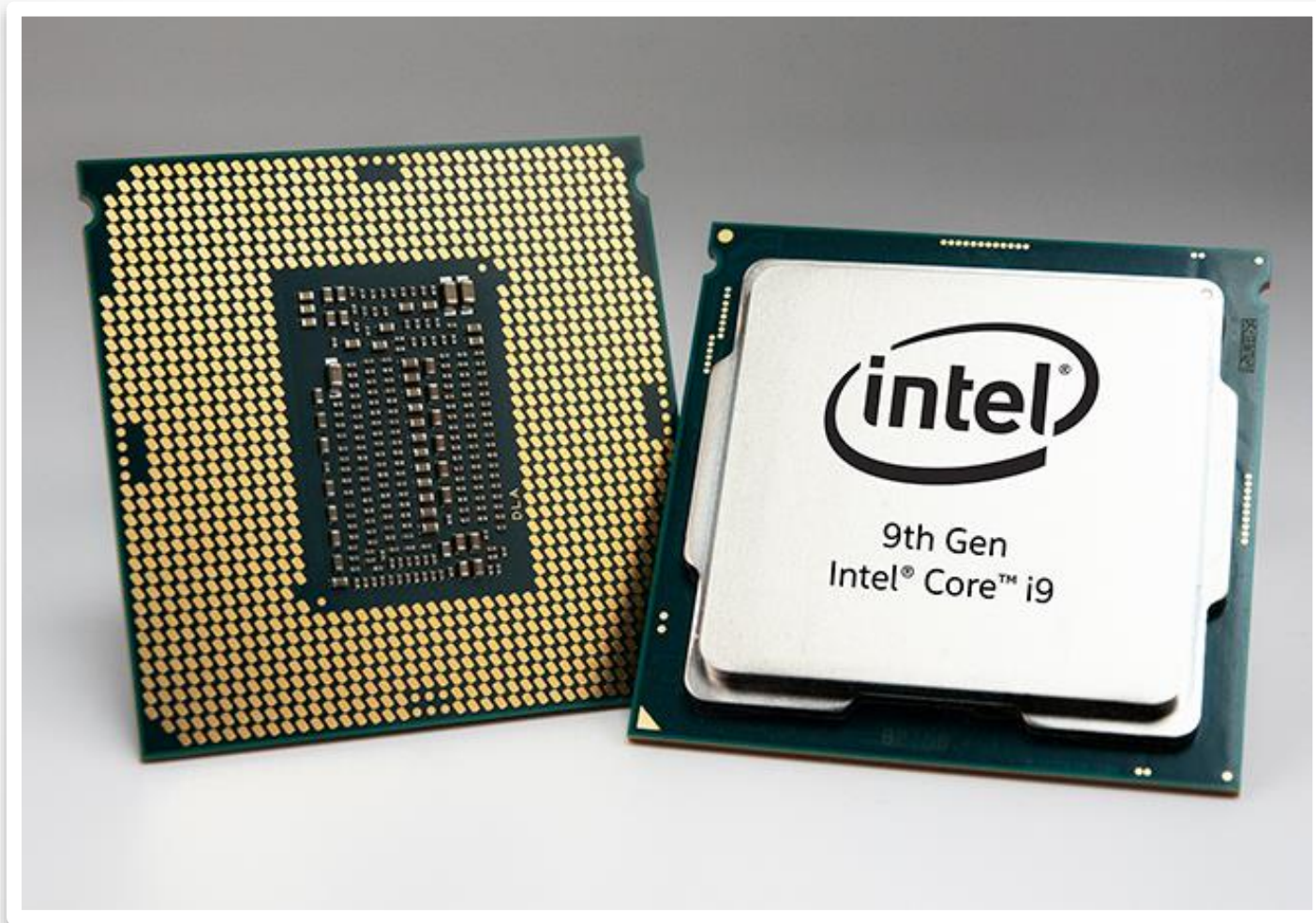
Σχηματικό διάγραμμα μητρικής κάρτας PC



Πηγή : <https://en.wikipedia.org/wiki/Motherboard>



# Αρχιτεκτονική Η/Υ



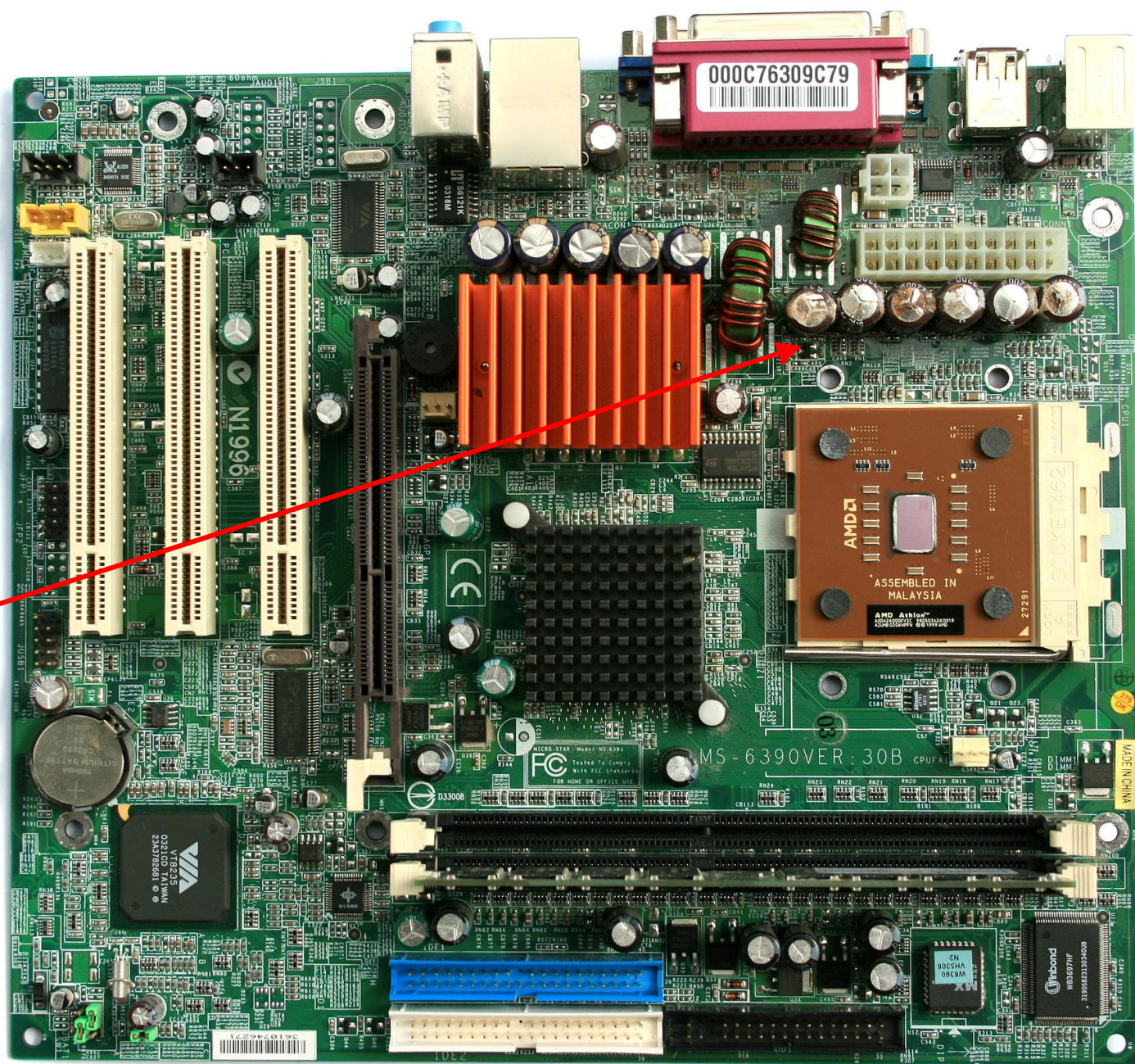
Εμπορικά διαθέσιμη (2019) CPU της εταιρίας Intel για PC





# Αρχιτεκτονική Η/Υ

μητρική κάρτα τύπου  
microATX για PC  
(παρατηρήστε τους  
σκασμένους  
πυκνωτές)



\*-ware

software

firmware

hardware

```
#include <stdio.h>
int main() {
    // printf() displays the string inside quotation
    printf("Hello, World!");
    return 0;
}
```





# Περί γλωσσών

Η γλώσσα επικοινωνίας σταδιακά αλλάζει από επίπεδο σε επίπεδο. Διαδοχικά επίπεδα επικοινωνούν - «συνομιλούν» μέσω αμοιβαία κατανοητών στοιχείων τα οποία ορίζουν μια «διεπαφή» (interface) υποβολής και ανταπόκρισης αιτημάτων



# Περί γλωσσών

Συμβολική γλώσσα  
(assembly language)  
για επεξεργαστή της  
οικογένειας x86-64

```
_fib:
    movl $1, %eax
.fib_loop:
    cmpl $1, %edi
    jbe .fib_done
    movl %eax, %ecx
    addl %ebx, %eax
    movl %ecx, %ebx
    subl $1, %edi
    jmp .fib_loop
.fib_done:
    ret
```

Γλώσσα μηχανής (Machine code)  
για επεξεργαστή τύπου 32-bit x86

```
8B542408 83FA0077 06B80000 0000C383
FA027706 B8010000 00C353BB 01000000
B9010000 008D0419 83FA0376 078BD989
C14AEBF1 5BC3
```

Κατώτερα επίπεδα γλωσσών προγραμματισμού  
([https://en.wikipedia.org/wiki/Low-level\\_programming\\_language](https://en.wikipedia.org/wiki/Low-level_programming_language))





# Περί γλωσσών

Συμβολικές γλώσσες Wolfram Mathematica,  
προγραμματισμού Matlab, κλπ.



python™



Scripting languages

Παράδειγμα κώδικα  
στη γλώσσα C

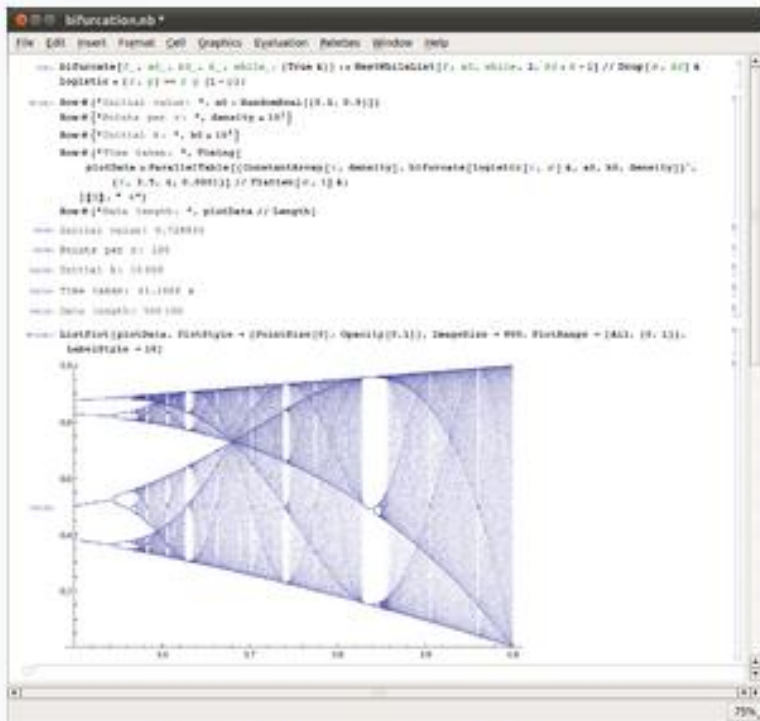
```
unsigned fib(unsigned n) {  
    if (!n)  
        return 0;  
    else if (n <= 2)  
        return 1;  
    else {  
        unsigned a, c;  
        for (a = c = 1; ; --n) {  
            c += a;  
            if (n <= 2) return c;  
            a = c - a;  
        }  
    }  
}
```

Ανώτερα επίπεδα γλωσσών προγραμματισμού  
([https://en.wikipedia.org/wiki/High-level\\_programming\\_language](https://en.wikipedia.org/wiki/High-level_programming_language))

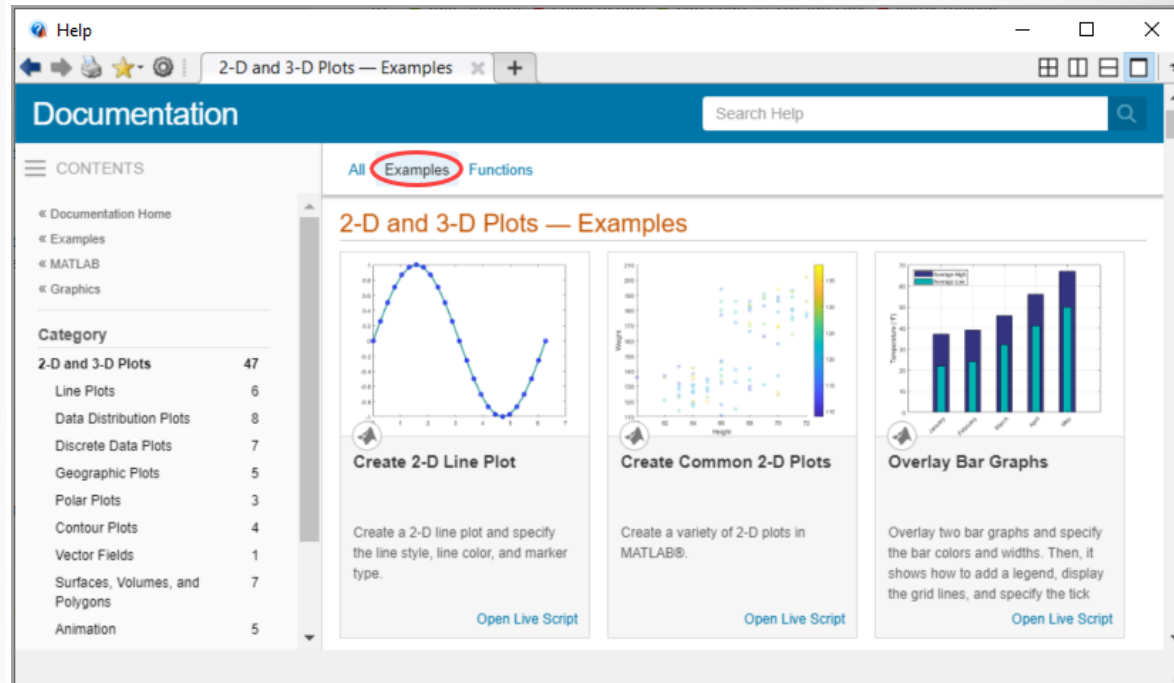


# Περί γλωσσών

## Wolfram Mathematica



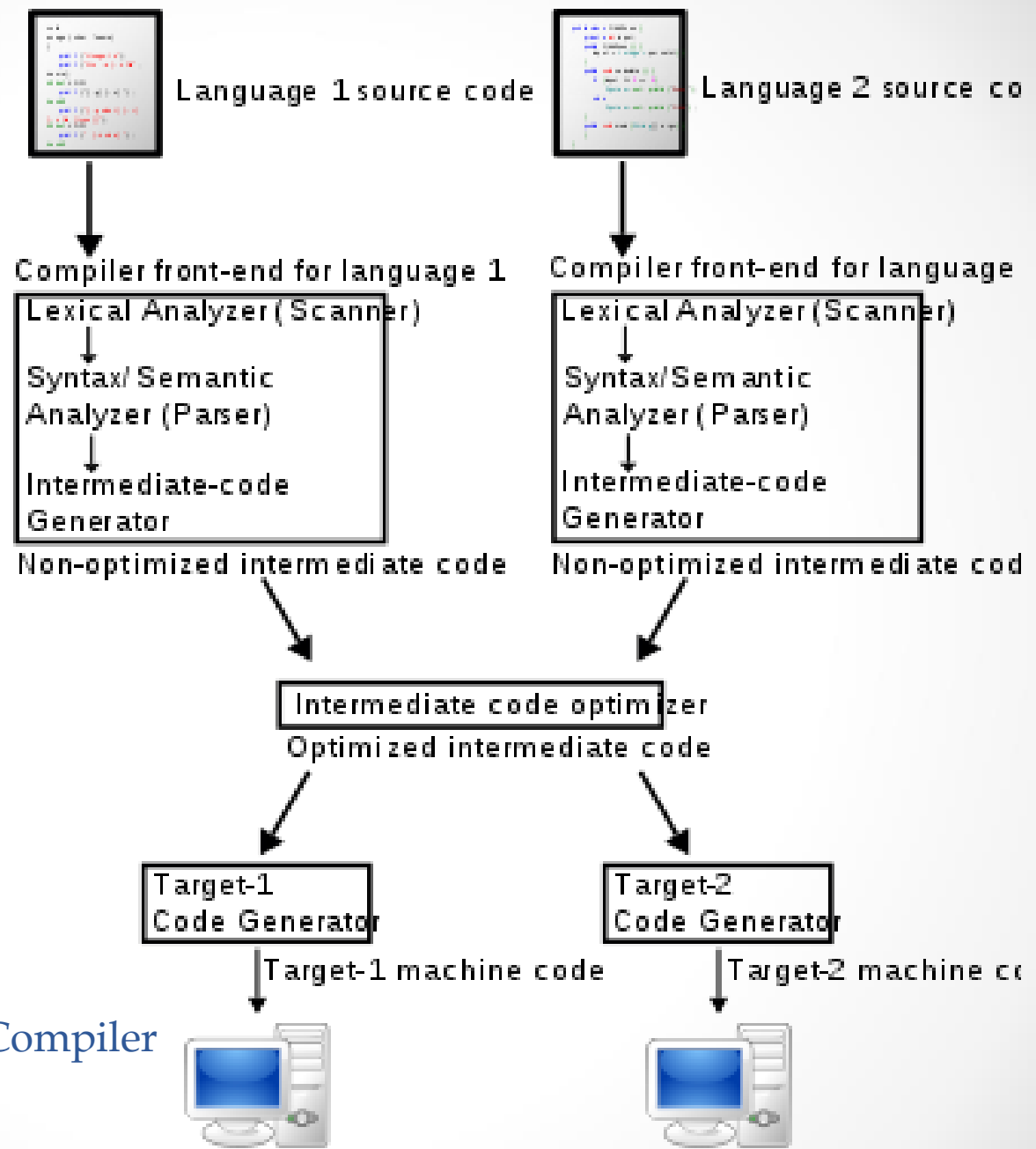
Mathematica 8.0.0 Linux frontend



Συμβολικές γλώσσες προγραμματισμού



# Μεταφραστές



<https://en.wikipedia.org/wiki/Compiler>

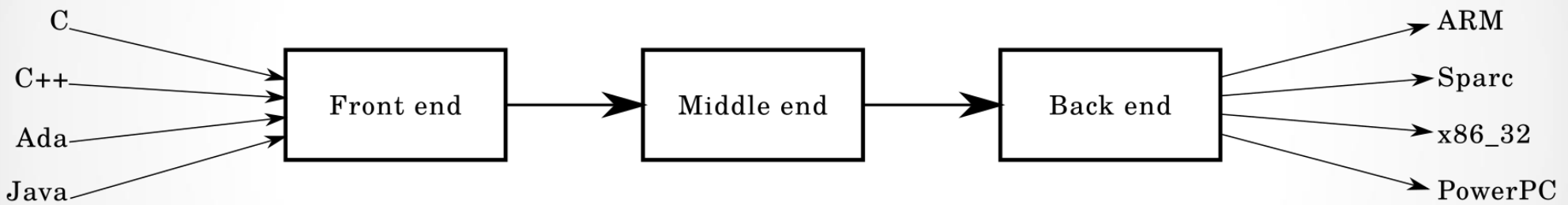


Τμήμα Μαθηματικών

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης



# Μεταφραστές



# Μεταφραστές

The screenshot shows the Dev-C++ 5.11 application window. The 'About Dev-C++' dialog box is open, displaying the following information:

- Copyright (c) Bloodshed Software** Version 5.11  
Build time: April 27 2015 - 17:11
- GNU General Public License**  
GNU GENERAL PUBLIC LICENSE  
Version 2, June 1991  
Copyright (C) 1989, 1991 Free Software Foundation, Inc.  
675 Mass Ave, Cambridge, MA 02139, USA
- Contact Information**
  - > 4.9.9.2
  - Repository: <https://sf.net/projects/orwelldevcpp/>
  - Subreddit: <http://www.reddit.com/r/devcpp/>
  - Blog: <http://orwelldevcpp.blogspot.com>
- Donate** (button)
- Compilers**
  - MinGW Compiler Website: <http://www.mingw.org>
  - TDM-GCC Compiler Website: <http://tdm-gcc.tdragon.net/>
- <= 4.9.9.2 (old stuff)**
  - Bloodshed Software Website: <http://www.bloodshed.net>
  - Dev-C++ discussion forums: <http://www.bloodshed.net/forum/>
  - Dev-C++ users mailing list: <http://www.bloodshed.net/devcpp-ml.html>
  - Authors e-mail: <http://www.bloodshed.net/dev/>

Buttons at the bottom of the 'About' dialog: Check for updates, Authors, OK.

The 'Information' dialog box is also open, showing the following authors:

- Authors:**
  - Post-4.9.9.2 development: Johan Mes
  - Development: Colin Laplace, Mike Berg, Hongli Lai, Yiannis Mandravellos
  - Contributors: Peter Schraut, Marek Januszewski, Anonymous
  - MinGW compiler system: Mumit Khan, J.J. van der Heijden, Colin Hendrix and GNU developers
  - Splash screen and association icons: Matthijs Crieleard
  - New Look theme: Gerard Caufield
  - Gnome icons: Gnome designers
  - Blue theme: Thomas Thron

Button at the bottom of the 'Information' dialog: OK.



# Μεταφραστές VS διερμηνευτές

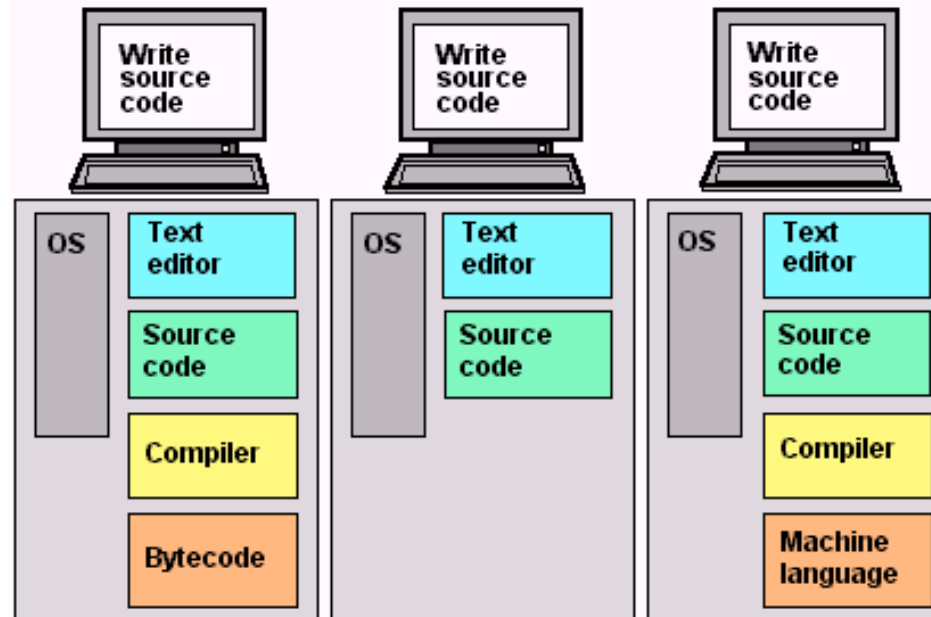
From Computer Desktop Encyclopedia  
© 2000 The Computer Language Co. Inc.

## Create & Modify

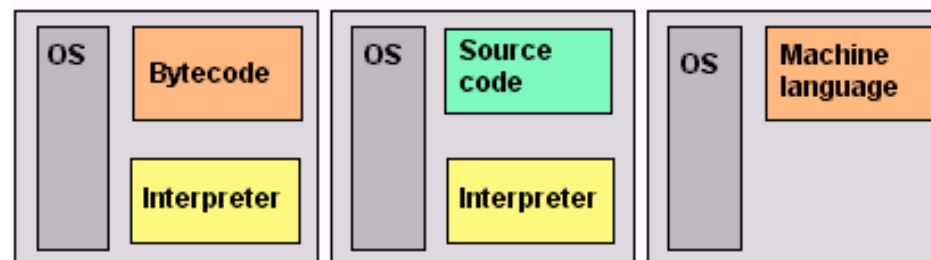
Java, Visual Basic  
(interpreted)

dBASE, BASIC, etc.  
(interpreted)

C, C++, COBOL, etc.  
(compiled)



## Run



# Ερωτήσεις & Απαντήσεις

