

# Εισαγωγή στην Πληροφορική

Ανδρέας Παπασαλούρος  
andrapas@aegean.gr

# Σχετικά με το μάθημα (1)

- Ώρες Μαθήματος:
  - Δευτέρα 18-20, Κτίριο Εμπορικής, Αιθ. Α1.
  - Τρίτη 9-11, Κτίριο Εμπορικής, Αιθ. Α1.
- Εργαστήριο: Παρασκευή 9-13
- κκ. Τσαγγάρης και Παπαλουκάς.
- Ύλη Εργαστηρίου:
  - Χρήση του Υπολογιστή
  - Περιβάλλον ανάπτυξης
  - Εισαγωγή στην γλώσσα προγραμματισμού C
- Βαθμολογία
  - Μια σειρά ασκήσεων που θα παραδίδονται κατά τη διάρκεια του εξαμήνου (20%)
  - Τελικές γραπτές εξετάσεις (80%)

# Σχετικά με το μάθημα (2)

Σελίδα του μαθήματος (σύντομα διαθέσιμη)

<http://www.samos.aegean.gr/math/andrapas/courses/ics/>

- Οι παρουσιάσεις του μαθήματος θα αναρτώνται στη σελίδα σε μορφή διαφανειών καθώς και οι ασκήσεις προς παράδοση.
- Βιβλίο μαθήματος: Eric S. Roberts, *Η Τέχνη και Επιστήμη της C, Μια εισαγωγή στην Επιστήμη των Υπολογιστών*, Κλειδάριθμος.
- Ώρες Επικοινωνίας με τους φοιτητές:  
Τετάρτη και Πέμπτη 12-14.
- Επικοινωνία μέσω e-mail:  
andrapas [at] aegean [dot] gr

# Σκοπός του Μαθήματος

Μια εισαγωγή στην Επιστήμη των  
Υπολογιστών

Μια εισαγωγή στον Προγραμματισμό

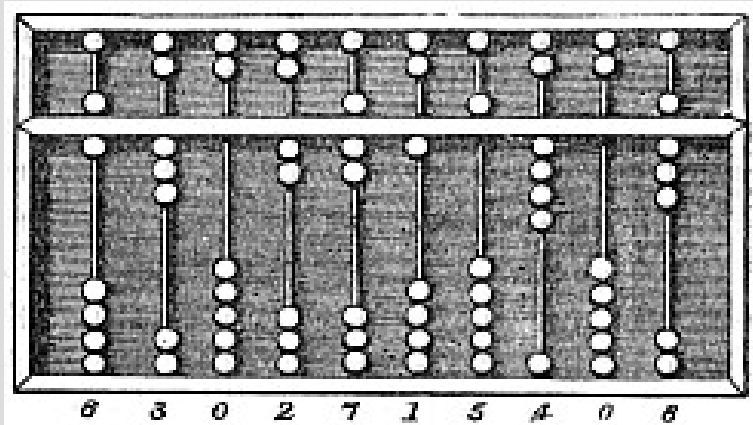
# Εφαρμογές των Υπολογιστών

- Ιατρικές
- Στρατιωτικές
- Οικονομικές - επιχειρησιακές  
(business/enterprise information systems)
- Διασκέδαση/ψυχαγωγία
- Τέχνη
- Επικοινωνία
- ...

# Μαθηματικά και Υπολογιστές

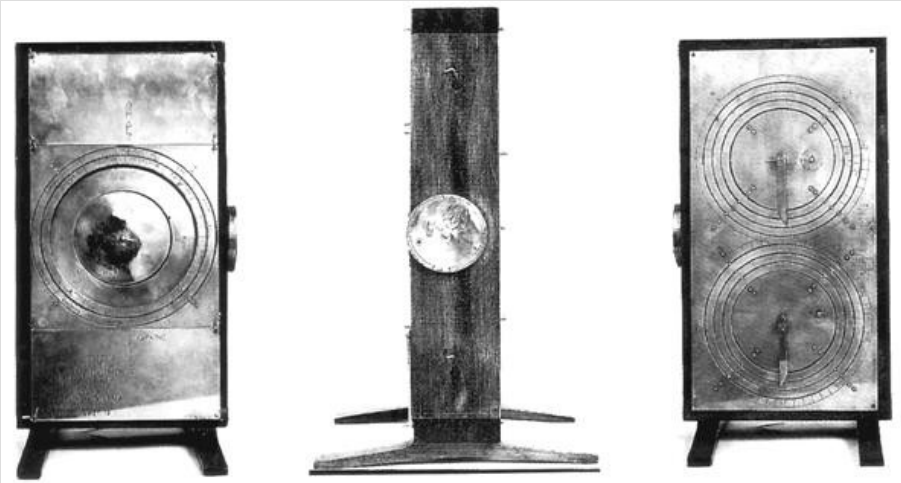
- Η μαθηματική θεμελίωση της Επιστήμης των Υπολογιστών
  - Λογική
  - Θεωρία Αριθμών
  - Θεωρία Γραφημάτων
  - Θεωρία Τύπων
- Η χρήση της Επιστήμης των Υπολογιστών στα Μαθηματικά

# Άβακας (2η Χιλιετηρίδα π.Χ.)



- Εφευρέθηκε στην Βαβυλώνα
- Επέτρεπε την διενέργεια των τεσσάρων αριθμητικών πράξεων με ταχύτητα
- Διάφορες παραλλαγές εντοπίζονται σε διαφορετικούς πολιτισμούς (Κίνα, Ιαπωνία, κ.λπ.).

# Μηχανισμός των Αντικυθήρων (5ος αι. π.Χ.)



Αρχαιολογικό Μουσείο Αθηνών

- Υπολόγιζε:
  - Ετήσια κίνηση του ήλιου στον ζωδιακό κύκλο
  - Ανατολές/ δύσεις λαμπρών άστρων και αστερισμών
- Αποτελούνταν από ένα σύστημα 32 οδοντωτών τροχών

Κέντρο Διάδοσης Επιστημών και  
Μουσείο Τεχνολογίας

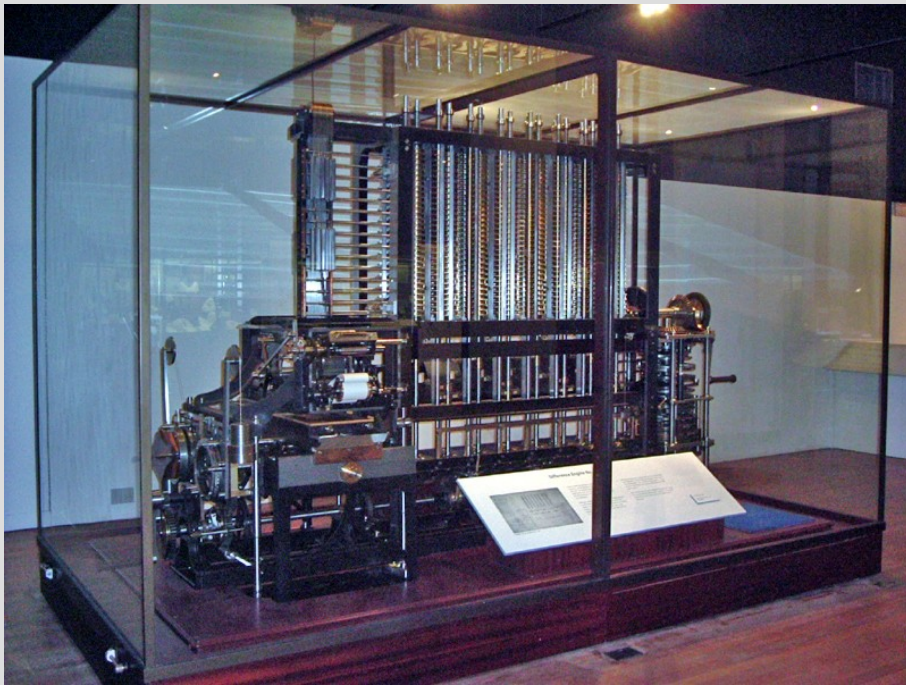
[http://www.tmth.edu.gr/el/kiosks/computers/history/comp\\_a2.html](http://www.tmth.edu.gr/el/kiosks/computers/history/comp_a2.html)



# Σύγχρονες Μηχανές (16ος – 18ος αι.)

- Wilhelm Schickard
- Blaise Pascal
- Gottfried Leibniz

# Μηχανές του Babbage



- **Μηχανή Διαφορών:**  
Παρήγαγε πίνακες  
μαθηματικών  
συναρτήσεων
- **Αναλυτική Μηχανή:**
  - Προγραμματιζόμενη με  
χρήση διάτρητων  
καρτών
  - Ποτέ δεν υλοποιήθηκε
  - Το πρώτο πρόγραμμα  
για την αναλυτική  
μηχανή από την Ada  
Lovelace

# Μηχανή του Jacquard



- Αυτοματοποίηση της διαδικασίας ύφανσης σε αργαλειό
- *Προγραμματιζόμενη μηχανή*

# Εφαρμογές διάτρητων καρτών



Τεχνική για την μηχανική  
αποτύπωσης της  
πληροφορίας.  
Μηχανικό πιάνο  
(πιανόλα), Λατέρνα (!)

# Η μηχανή του Hollerith

From Computer Desktop Encyclopedia  
Reproduced with permission.  
From Computer Desktop Encyclopedia  
Reproduced with permission. Center  
From Computer Desktop Encyclopedia  
Reproduced with permission.  
© 1996 Scientific American Magazine



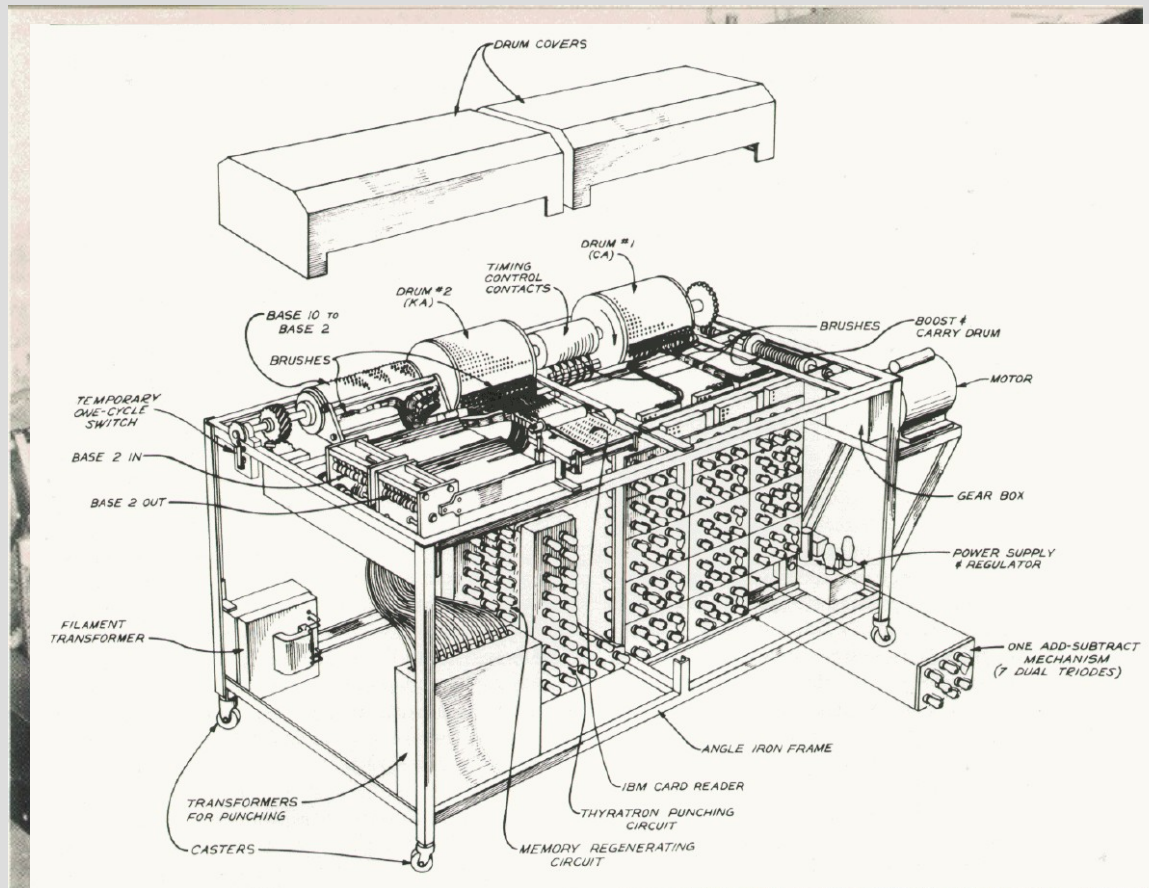
- Χρησιμοποιήθηκε για την αυτοματοποίηση της επεξεργασίας δεδομένων κατά την απογραφή του 1890 στις ΗΠΑ.
- Τα δεδομένα αποτυπώνονταν σε διάτρητες κάρτες.
- Ο Hollerith ίδρυσε την εταιρεία International Business Machines (IBM)



# Από τις λυχνίες κενού στους σύγχρονους υπολογιστές

- Πρώτη γενιά: Χρήση λυχνιών κενού (Atanasoff, 1939)
- Δεύτερη γενιά: Χρήση τρανζίστορ (IBM 7090, 1957)
- Τρίτη γενιά: Χρήση ολοκληρωμένων κυκλωμάτων (IBM 360, 1964)
- Τέταρτη γενιά: Χρήση κυκλωμάτων μεγάλης κλίμακας ολοκλήρωσης και μικροεπεξεργαστών,
- Μικροϋπολογιστές και προσωπικοί υπολογιστές (1975)

# Μηχανή του Atanasoff



- Ο πρώτος ψηφιακός υπολογιστής
- Βασιζόταν στην τεχνολογία των λυχνιών κενού
- Προοριζόταν για την επίλυση μικρών συστημάτων εξισώσεων

# Η λυχνία κενού

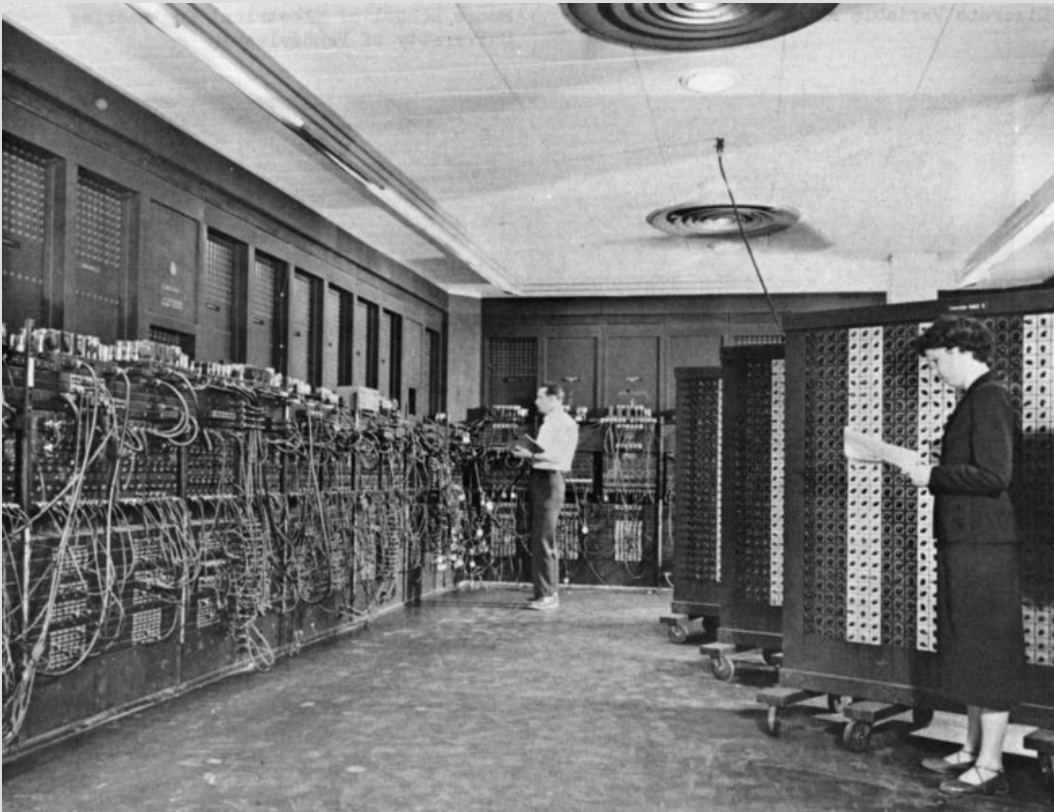


- Βασικό στοιχείο της μονάδας υπολογισμού (κεντρικής μονάδας)
- ενός υπολογιστή.
- Αντικαταστάθηκε από το τρανζίστορ και το ολοκληρωμένο κύκλωμα.

<http://www.cs.virginia.edu/brochure/museum.html>

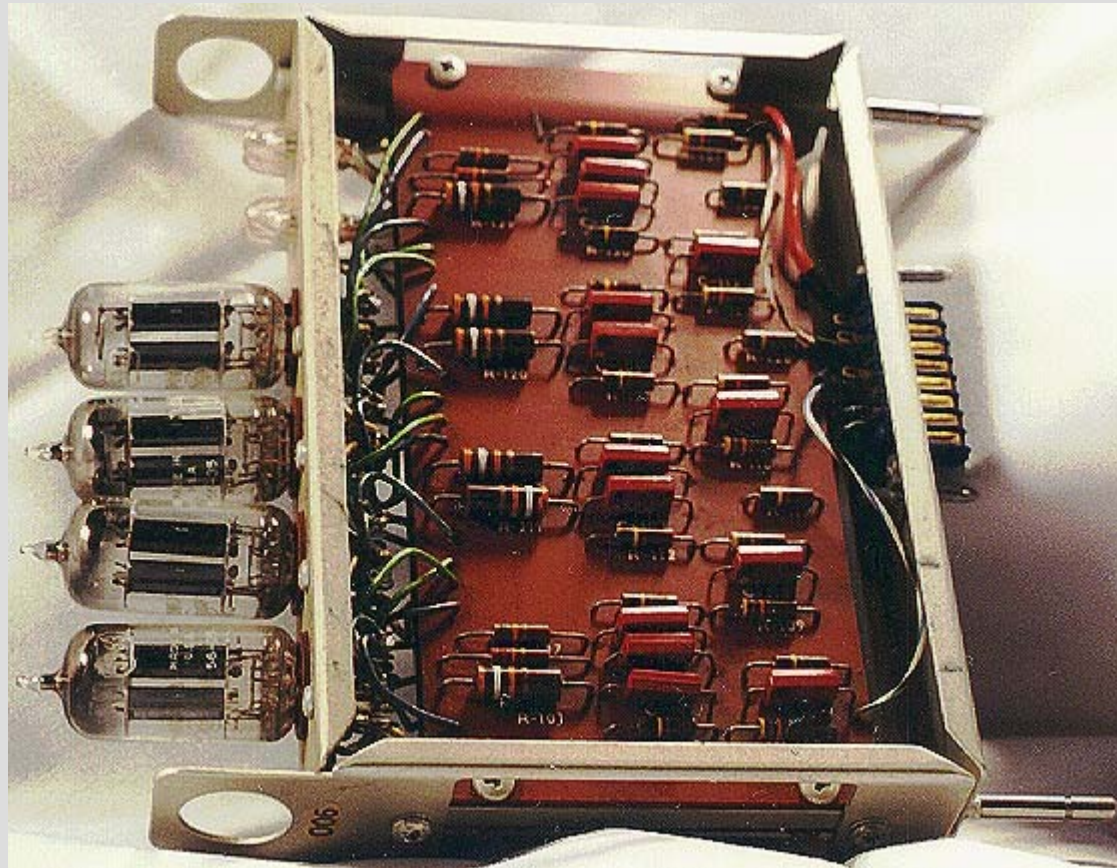


# ENIAC (1946)



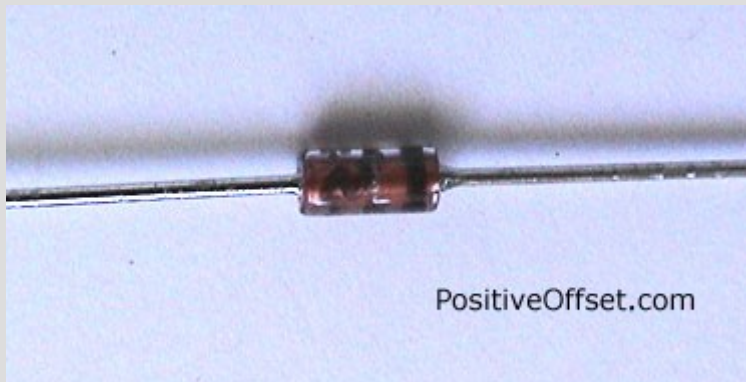
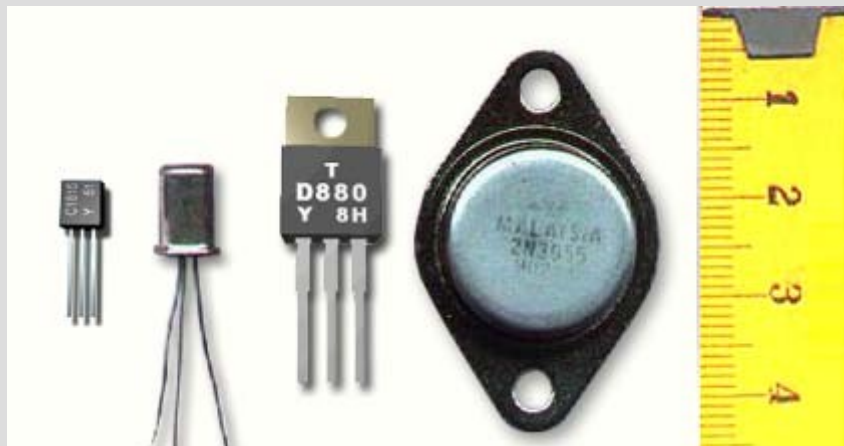
- Προοριζόταν για την επίλυση προβλημάτων βαλλιστικής (υπολογισμός πινάκων βολών)
- Προγραμματιζόταν με την τοποθέτηση καλωδίων σε μια διάτρητη επιφάνεια (patch panel).

# Παράδειγμα χρήσης λυχνίας



Απαριθμητής τεσσάρων bit

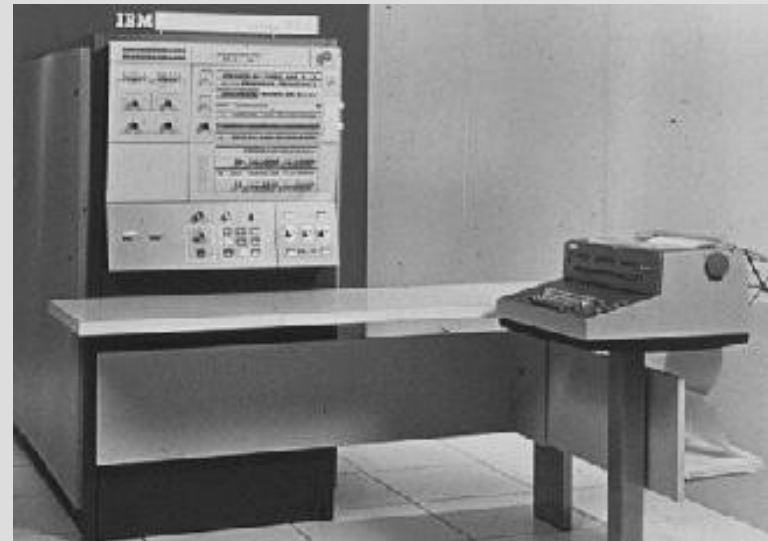
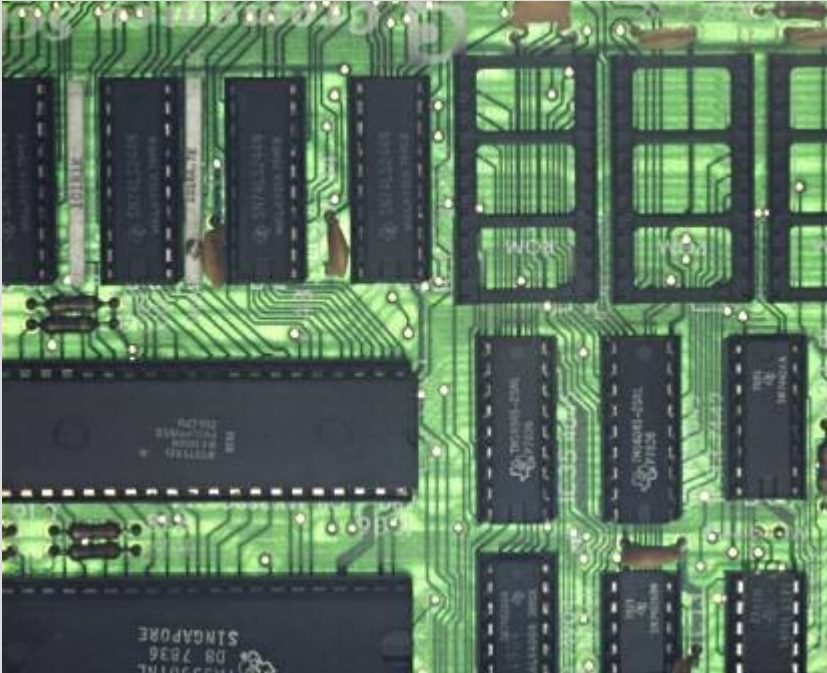
# Υπολογιστές με τρανζίστορ



IBM 7090



# Υπολογιστές με ολοκληρωμένα κυκλώματα



IBM 360

# Ολοκληρωμένα κυκλώματα μεγάλης κλίμακας ολοκλήρωσης



<http://www.intel.com>



Apple Mac Pro

# Χαρακτηριστικά σύγχρονων υπολογιστικών συστημάτων

- Παραλληλία
- Δικτύωση – κατανεμημένος υπολογισμός
- Ασφάλεια
- Προηγμένη επικοινωνία ανθρώπου-υπολογιστή
- Ενσωμάτωση σε συσκευές

# Η Επιστήμη των Υπολογιστών

- Ο υπολογιστής είναι μια μηχανή που εκτελεί προγραμματισμένους υπολογισμούς.
- Ένας υπολογιστής συνίσταται από υλικό και λογισμικό.
- Η Επιστήμη των Υπολογιστών ασχολείται με τη μελέτη της φύσης αυτών των υπολογισμών.
- Εστιάζει περισσότερο στο Λογισμικό



# Υπολογισμός και αλγόριθμοι

- Αλγόριθμος: Μια **στρατηγική** για την επίλυση ενός **προβλήματος** με τα παρακάτω χαρακτηριστικά:
  - **Σαφής** και απερίφραστα ορισμένος
  - **Αποτελεσματικός**
  - **Πεπερασμένος**
- Η λέξη αλγόριθμος προέρχεται από το όνομα ενός άραβα μαθηματικού του 9ου αι.

# Παράδειγμα αλγορίθμου

**Πρόβλημα:** Ένας ταμίας θέλει να υπολογίσει τα ρέστα που πρέπει να δώσει σε μια πληρωμή, ώστε ο αριθμός νομισμάτων που θα επιστρέψει να είναι ελάχιστος.

**Είσοδος:** Η αξία της αγοράς  
Το ποσό πληρωμής.

**Έξοδος:** Μια λίστα από νομίσματα που επιστρέφονται ως ρέστα.

**Υποθέτουμε ότι:**

- το ποσό πληρωμής είναι μεγαλύτερο ή ίσο από την αξία της αγοράς
- ο ταμίας έχει απεριόριστο αριθμό νομισμάτων οποιασδήποτε αξίας (αν δεν έχει, πετάγεται “δίπλα”).



# Περιγραφή του αλγορίθμου

1. Αρχικά η διαφορά είναι ίση με το ποσό πληρωμής **μείον** την **αξία της αγοράς**.
2. **Όσο** η διαφορά είναι μεγαλύτερη από το μηδέν, **επανάλαβε** τα βήματα 3 έως 4.
- 3 Βρες το **μεγαλύτερο νόμισμα** με αξία **μικρότερη ή ίση** από την **διαφορά**.
4. **Πρόσθεσε** το νόμισμα στη λίστα με τα ρέστα και **αφαίρεσε** την αξία του από τη διαφορά.

# Παράδειγμα εκτέλεσης του αλγορίθμου

1. Αρχικά η διαφορά είναι ίση με το ποσό πληρωμής μείον την αξία της αγοράς.
2. **Όσο** η διαφορά είναι μεγαλύτερη από το μηδέν, **επανάλαβε** τα βήματα 3 έως 4.
3. Βρες το **μεγαλύτερο νόμισμα** με αξία **μικρότερη ή ίση** από την **διαφορά**.
4. **Πρόσθεσε** το νόμισμα στη λίστα με τα ρέστα και **αφαίρεσε** την αξία του από τη διαφορά.

Αρχικά:

Αξία αγοράς 5,25 €  
Ποσό πληρωμής: 20 €  
Διαφορά: 14,75 €

1η επανάληψη

Νόμισμα 10€, διαφορά 4,75 €

2η επανάληψη

Νόμισμα 2 €, διαφορά 2,75€

3η επανάληψη

Νόμισμα 2€, διαφορά 0,75€

4η επανάληψη

Νόμισμα 0,50 €, διαφορά 0,25€

5η επανάληψη

Νόμισμα 0,20 €, διαφορά 0,05 €

6η επανάληψη

Νόμισμα 0,05€, διαφορά 0€.

# Συζήτηση

- Συνιστά η παραπάνω περιγραφή έναν αλγόριθμο; (σαφής, αποτελεσματική και πλήρης);
- *Είναι ο αλγόριθμος ορθός;*
- *Τερματίζει;*
- *Μετά από πόσα βήματα (χρόνο) τερματίζει;*
- *Είναι ο αλγόριθμος αυτός **βέλτιστος**, δηλαδή λύνει το πρόβλημα με ελάχιστο αριθμό βημάτων;*

Τα παραπάνω τρία ερωτήματα είναι θεμελιώδη για την επιστήμη των υπολογιστών και ιδιαίτερα την μελέτη των αλγορίθμων.

# Περιγραφή αλγορίθμων

Φυσική γλώσσα  
Διαγράμματα ροής  
Ψευδοκώδικας  
Γλώσσες προγραμματισμού



# Ο ρόλος του Alan Turing στην Επιστήμη των Υπολογιστών

- Τυποποίηση των εννοιών του αλγορίθμου και του υπολογισμού.
- Εισήγαγε την αφηρημένη μηχανή υπολογισμού (Turing machine).
- Σχεδίαση τον υπολογιστή ACE (Automatic Computing Engine).
- Πρότεινε το περίφημο Turing test για τον καθορισμό της Τεχνητής Νοημοσύνης (Artificial Intelligence - AI).
- Σχεδίασε μια μηχανή αποκρυπτογράφησης του κώδικα Enigma (Turing-Welchman bombe) κατά τον Β΄ Παγκόσμιο Πόλεμο.



# Γλώσσες Προγραμματισμού

Οι γλώσσες προγραμματισμού επιτρέπουν την δημιουργία προγραμμάτων για την εκτέλεση αλγορίθμων από υπολογιστικά συστήματα.

Η Επιστήμη των Υπολογιστών δεν εξαντλείται στον προγραμματισμό και τις Γλώσσες Προγραμματισμού.

Ένα μεγάλο σύνολο γλωσσών προγραμματισμού είναι διαθέσιμες.

Στο πλαίσιο του μαθήματος θα ασχοληθούμε με την γλώσσα προγραμματισμού C

# Παραδείγματα προβλημάτων που λύνονται από υπολογιστές

Πρόβλημα 1: Διαπίστωση του αν ένα στοιχείο  
είναι μέλος ενός συνόλου (αναζήτηση).

Πρόβλημα 2: Πρόβλημα εύρεσης του μεγαλύτερου  
και του μικρότερου στοιχείου σε ένα σύνολο  
αριθμών;

Πρόβλημα 3: Πολλαπλασιασμός πινάκων

Πρόβλημα 4: Επίλυση εξίσωσης πρώτου βαθμού  
$$ax + \beta = 0$$

Πρόβλημα 5: Ταξινόμηση ενός συνόλου  
αταξινομήτων αριθμών.

# Βασικά στοιχεία αλγορίθμων

Βήματα (εντολές)

Σημεία απόφασης (**Αν ... τότε ...**)

Επαναλήψεις (**για όσο ... επανάλαβε ...**)

# Τα όρια του υπολογισμού

- Υπάρχουν προβλήματα που έχει αποδειχτεί ότι δεν είναι δυνατόν να επιλυθούν (τουλάχιστον ακριβώς) από τους υπολογιστές (*undecidable*):
  - Halting problem, το δέκατο πρόβλημα του Hilbert (Θεώρημα του Matiyasevich).
- Επιπλέον, υπάρχουν προβλήματα που ο χρόνος επίλυσής τους είναι τεράστιος, για μη τετριμμένο μέγεθος του προβλήματος (*intractable*).
- Παράδειγμα: Το πρόβλημα του περιοδεύοντος πωλητή (TSP).

# Παράδειγμα “δύσκολου” προβλήματος

- Χρωματισμός γράφων (graph coloring)
- Είναι δυνατός ο χρωματισμός των κόμβων του γράφου με 3 χρώματα;
- Ένας χρωματισμός είναι έγκυρος αν κανένα ζευγάρι κόμβων που συνδέονται με μια ακμή δεν έχουν το ίδιο χρώμα.
- Η λύση του παιχνιδιού Sudoku είναι δυνατόν να αναχθεί σε ένα πρόβλημα χρωματισμού γράφων
- Ο καθορισμός του προγράμματος διδασκαλίας σε ένα σχολείο επίσης ανάγεται στο πρόβλημα χρωματισμού γράφων.

