



**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΑΛΓΟΡΙΘΜΟΙ  
ΚΑΙ  
ΠΟΛΥΠΛΟΚΟΤΗΤΑ**

**Β. ΖΗΣΙΜΟΠΟΥΛΟΣ**

**ΑΘΗΝΑ  
2011**

# Ερώτηση

- Η λέξη “αλγόριθμος” προέρχεται από
  - Τη λέξη άλγος;
  - Τον Muhamed Ibn Musa Al Khov Warizmi
  - Κανένα από τα 2



\*\*\*\* [1] Th. H. Cormen, CH. E. Leiserson, R. L. Rivest and C. Stein, Introduction to algorithms, MIT-Press, 1991 (1<sup>er</sup> edition), 2001 (2<sup>nd</sup> edition), 2009 (3<sup>rd</sup> edition) + translation in greek (2007). \*\*\*

<http://mitpress.mit.edu/algorithms/>

\*\*\* [2] S. Dasgupta, C. H. Papadimitriou & U. V. Vazirani, Algorithms, McGraw-Hill, 2008

\*\*\* [3] Jon Kleinberg & Eva Tardos, Algorithm Design, Addison – Wesley, 2006

[4] R. Sedgewick, Algorithms in C, Addison – Wesley, 2nd ed., 1998.

[5] S. S. Skiena, The algorithm design manuel, Springer – Verlag, 1998.

# Εξέταση

1. Π: Πρόοδος 25%
  2. Ε: 3 Εργασίες 15%
  3. Γ: Γραπτή εξέταση: 60%
- Τελικός βαθμός:  $0.25\Pi+0.15E+0.60\Gamma$

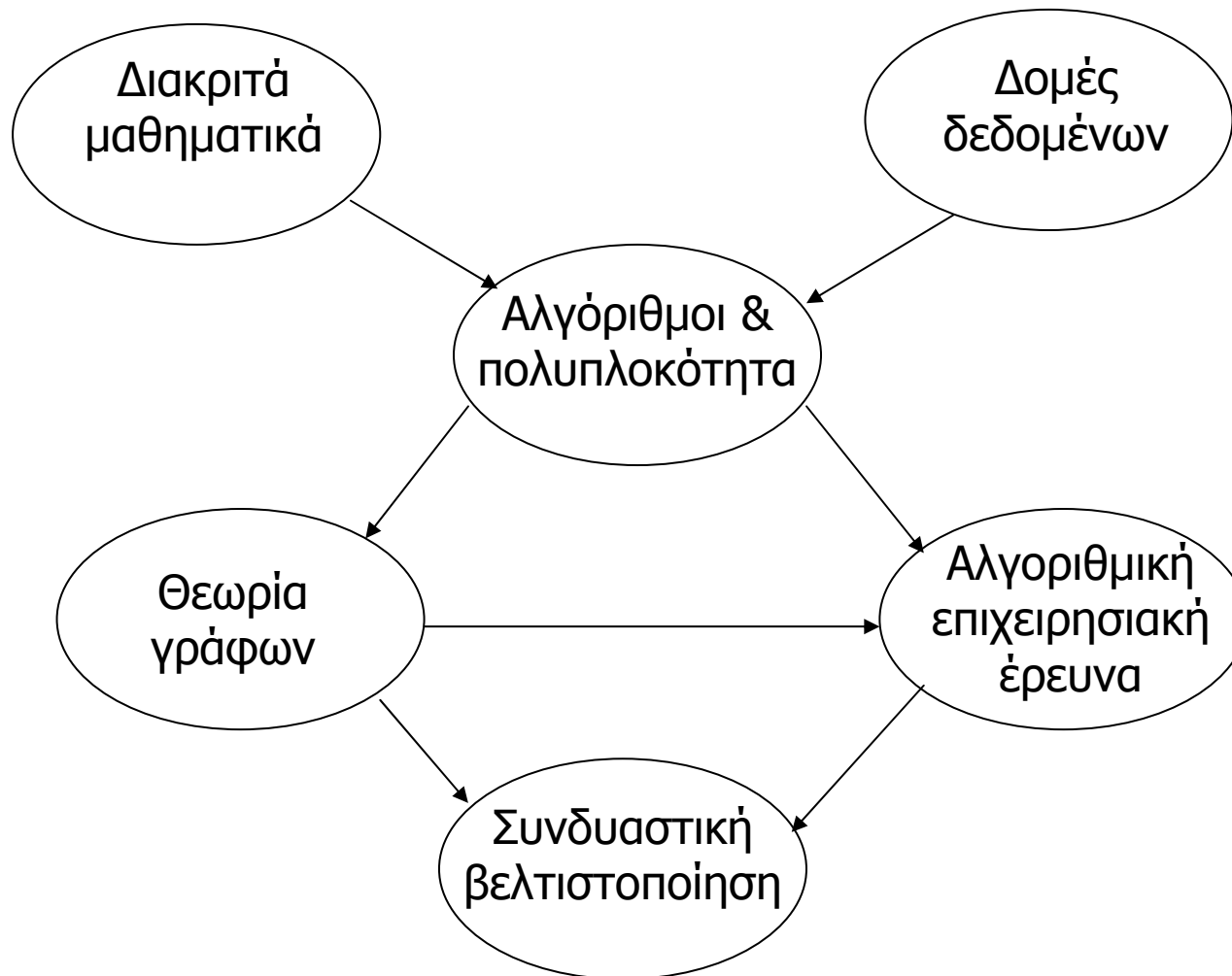
## Πληροφορίες

- [www.di.uoa.gr/~vassilis/](http://www.di.uoa.gr/~vassilis/)

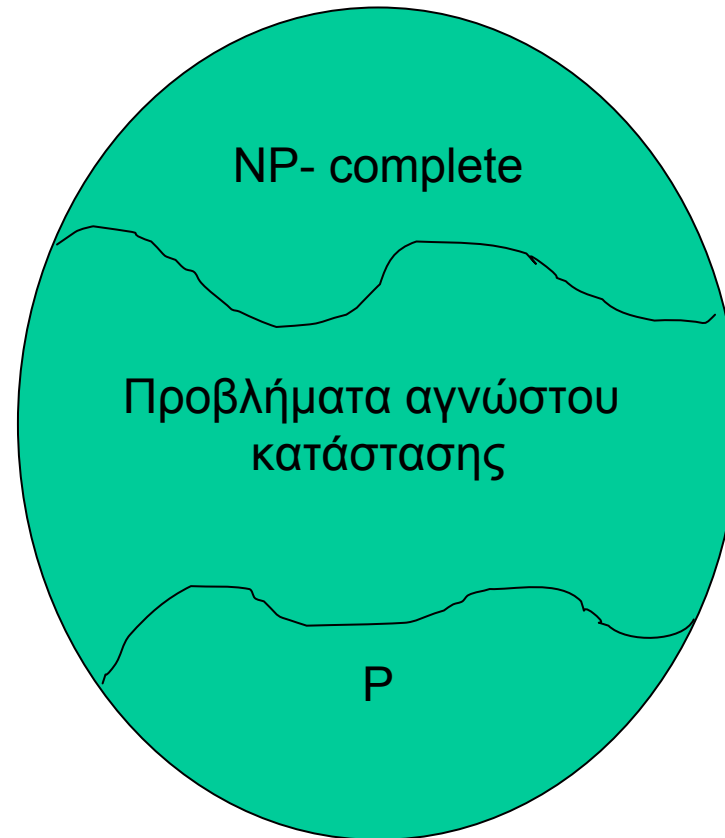
Announcements (for student) → Αλγόριθμοι και Πολυπλοκότητα

Teaching → Algorithms and Complexity

# Ροές Μαθημάτων



# Τα προβλήματα



NP

- Αποτίμηση της επίδοσης ενός Αλγορίθμου;  
\*\*\*
- Σύγκριση 2 αλγορίθμων που επιλύουν το ίδιο πρόβλημα; \*\*\*
- Ανάπτυξη ενός «καλού» αλγορίθμου για ένα πρόβλημα; \*\*\*

# Η έννοια του αλγορίθμου

➤ Γενικά, ένας αλγόριθμος επεξεργάζεται  
πληροφορίες + ...

\*\*\*

---

# Κάθε αλγόριθμος είναι προγραμματίσιμος?

πρόγραμμα  $\rightarrow$  αλγόριθμος

αλγόριθμος  $\overset{?}{\rightarrow}$  πρόγραμμα

Για τη συνέχεια :

	αλγόριθμος
	↓↑
	πρόγραμμα

## Ψευδογλώσσα

- Πίνακας με ονόματα σε αλφαβητική σειρά (όπως ο τηλεφωνικός κατάλογος)
- Αλγόριθμος (αναζητούμενο στοιχείο: **X**)

repeat

**Σύγκριση** του αναζητούμενου στοιχείου με το μεσαίο  
στοιχείο

**Αν είναι μικρότερο** τότε συνεχίζουμε την

αναζήτηση στο πρώτο ήμισυ

**διαφορετικά** συνεχίζουμε στο δεύτερο

until (x=μεσαίο στοιχείο ή πίνακας κενός)

# Απόδοση αλγορίθμου

• Ένα πρόγραμμα χρήσιμο

→ σε «λογικό» χρόνο

→ «λογικό» χώρο μνήμης

↓ Απόδοση αλγορίθμου

Πολυπλοκότητα αλγορίθμου

➤ χρόνος εκτέλεσης

➤ απαιτούμενη μνήμη

## Πρόβλημα 1

**Δεδομένα:**  $n$  ακέραιοι,  $a[1], a[2], \dots, a[n]$

**Ζητούμενο:** Να ταξινομήσουμε τους ακεραίους κατά αύξουσα τάξη.

## Πρόβλημα 2

**Δεδομένα:**  $n$  αντικείμενα,  $c[i], w[i]$  και ένα σακκίδιο χωρητικότητας  $b$

**Ζητούμενο:** Επιλογή των χρησιμότερων αντικειμένων που χωράνε στο  $b$

# Η έννοια του στιγμιοτύπου

Πρόβλημα 1:  $n$  και οι συγκεκριμένοι ακέραιοι

Πρόβλημα 2:  $n, c(i), w(i), b$

# Η διάσταση του προβλήματος

Πρόβλημα 1:  $n$ , το πλήθος αριθμών

Πρόβλημα 2:  $n$ , το πλήθος αντικειμένων

Πρόβλημα 3:  $K \times K$  \*\*\*

# Η έννοια της πολυπλοκότητας:

- Αποτίμηση της αποδοτικότητας του αλγορίθμου σε ΧΡΟΝΟ και ΜΝΗΜΗ
  - Μονάδα μέτρησης (ουσιώδης πράξη: σταθερός χρόνος).
  - Συνάρτηση της διάστασης των δεδομένων.

## Τρεις τύποι πολυπλοκότητας

**Πρόβλημα:**

**Δεδομένα:** Ένας πίνακας  $S$  στοιχείων

$a_1, a_2, \dots, a_n$  και ένα στοιχείο  $x$ .

**Ζητούμενο:** το στοιχείο  $x$  είναι μέσα στο  $S$  και  
αν ΝΑΙ, σε ποία θέση;

## Αναζήτηση (S,x);

$i:=1;$

while  $i \neq n+1$  and  $a_i \neq x$  do  $i:=i+1;$

if  $i > n$  then Αναζήτηση **απέτυχε**

else στοιχείο **x** στη θέση **i**

# Τρεις τύποι πολυπλοκότητας

- Πολυπλοκότητα στη βέλτιστη των περιπτώσεων
- Πολυπλοκότητα στη χείριστη των περιπτώσεων
- Πολυπλοκότητα κατά μέσο όρο

## Το σύνολο δεδομένων διάστασης $n$ : $D_n$

$$1) C_{\beta\pi}(n) = \min \{ \text{κόστος}(d), d \in D_n \}$$

πολυπλοκότητα στη βέλτιστη περίπτωση

$$2) C_{\chi\pi}(n) = \max \{ \text{κόστος}(d), d \in D_n \}$$

πολυπλοκότητα στη χειρίστη περίπτωση

$$3) C_{\mu\text{όρο}}(n) = \sum_{d \in D_n} p(d) * \text{κόστος}(d)$$

όπου  $p(d)$  η πιθανότητα το δεδομένο  $d$  να είναι είσοδος του  $A$  (πολυπλοκότητα στη μέση περίπτωση)

⋮

Αριθμητικό γινόμενο δύο διανυσμάτων

$$A=(a_i) \text{ και } B=(b_i) \text{ του } \mathbb{R}^n, A \cdot B = \sum_{k=1}^n a_k \cdot b_k$$

## Αριθμητικό Γινόμενο (A, B:vectors)

sp:=0;

for k=1 to n do

    sp:=sp+( $a_k \cdot b_k$ );

Αριθμητικό Γινόμενο:=sp