

Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: **Σ. Ζάχος, Δ. Φωτάκης**
Επιμέλεια διαφανειών: **Δ. Φωτάκης**

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Οργανωτικά

- Διδάσκοντες: **Σ. Ζάχος, Δ. Φωτάκης**
 - Βοηθοί διδασκαλίας: **Σ. Δήμος, Μ. Επιτρόπου, Μ. Ζαμπετάκης**
Θ. Λιανέας, Α. Μάντης
 - 4 ώρες **θεωρία** (και ασκήσεις)
 - Δευτέρα 15:00 – 17:00, αμφ. 2, νέο κτ. ΣΗΜΜΥ
 - Πέμπτη 17:00 – 19:00, αμφ. 4, νέο κτ. ΣΗΜΜΥ
- Βαθμολογία:
 - 80% τελική εξέταση (ασκήσεις). Τουλάχιστον **3.5 στα 8**.
 - 15% **γραπτές ασκήσεις** (4 σειρές)
 - 15% **προγραμματιστικές ασκήσεις** (4 σειρές)
- Ιστοσελίδα: <http://www.corelab.ece.ntua.gr/courses/algorithms/>
- Forum: moodle.softlab.ntua.gr

Γραπτές Ασκήσεις

- **Ατομικές** γραπτές ασκήσεις:
 - 4 σειρές γραπτών ασκήσεων, **ανά 2^η εβδομάδα**.
 - Ανακοίνωση Δευτέρα, παράδοση επόμενη Δευτέρα (τουλάχιστον 2 εβδομάδες).
 - Εκφώνηση στο site (και στο moodle).
 - 1^η γραπτή: **Πέμπτη 5/12 → Πέμπτη 19/12**.
 - **Λύσεις:** συζήτηση στο μάθημα (ανακοίνωση σχεδίου λύσεων;)
 - Σημαντικές γιατί **προετοιμάζουν για τις εξετάσεις!**
 - Ένα από τα θέματα των εξετάσεων παρόμοια με γραπτές (ή λυμένες) ασκήσεις.
 - Οι «περυσινοί» μπορούν **«κρατήσουν» βαθμούς** ασκήσεων.

Προγραμματιστικές Ασκήσεις

- **Ατομικές** προγραμματιστικές ασκήσεις:
 - 4 σειρές προγραμματιστικών ασκήσεων, **ανά 2^η εβδομάδα**.
 - Ανακοίνωση Δευτέρα, εκφώνηση στο site (και στο moodle).
 - Παράδοση μεθεπόμενη Δευτέρα (2 εβδομάδες μετά), με **υποβολή** κώδικα) στον **grader**.
 - 1^η προγραμματ.: **Δευτέρα 9/12 → Δευτέρα 23/12**.
 - Όταν χρειάζεται, επίδειξη στο CoReLab (μετά από συνεννόηση).
 - Οι «περυσινοί» μπορούν **«κρατήσουν» βαθμούς** ασκήσεων.

Επικοινωνία

□ Σ. Ζάχος:

- ***E-mail:*** zachos@cs.ntua.gr
Τηλέφ: 210 7721646
Γραφείο: 1.1.15

□ Δ. Φωτάκης:

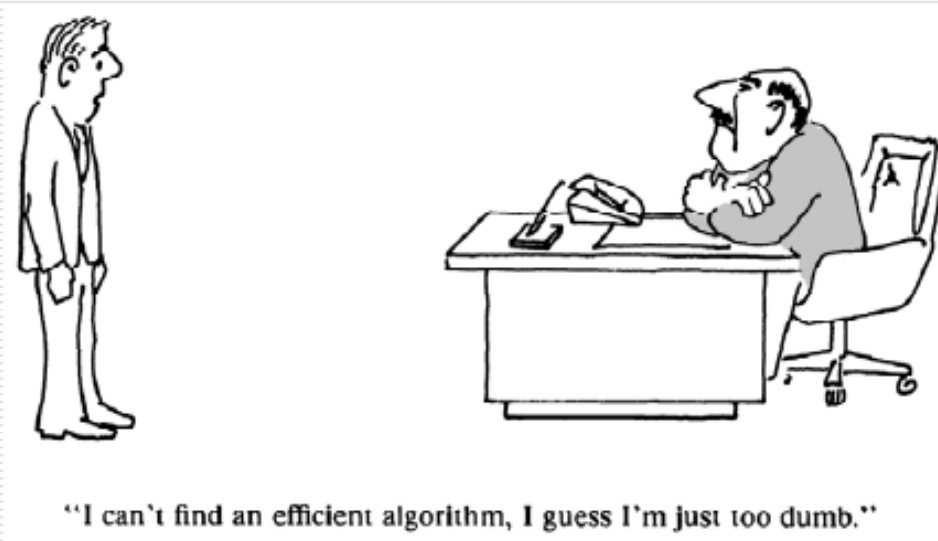
- ***E-mail:*** fotakis@cs.ntua.gr
Τηλεφ: 210 7724302
Γραφείο: 1.1.10
- ***Ώρες γραφείου:*** Δευτέρα 13:00–14:00 και Πέμπτη 14:00–16:00
στο **Corelab 1.1.3** (210 7723339) ή στο **1.1.10**.

Αλγόριθμοι και ...

- Γιατί κάποια **υπολογιστικά προβλήματα** είναι **εύκολο** και κάποια άλλα είναι **δύσκολο** να λυθούν από **υπολογιστικές μηχανές**.
 - «Εύκολα» προβλήματα: επιλύονται από **αλγόριθμους** που απαιτούν **εύλογη ποσότητα υπολογιστικών πόρων** (χρόνος, μνήμη, επικοινωνία).
 - Βασικές έννοιες και βασικές δομές δεδομένων.
 - Βασικές **τεχνικές** σχεδιασμού και ανάλυσης αλγορίθμων:
 - Διαίρει-και-Βασίλευε
 - Απληστία.
 - Δυναμικός προγραμματισμός
 - Εφαρμογή: **αλγόριθμοι γραφημάτων**
 - Ελάχιστο συνδετικό δέντρο
 - Συντομότερα μονοπάτια
 - Μέγιστη ροή

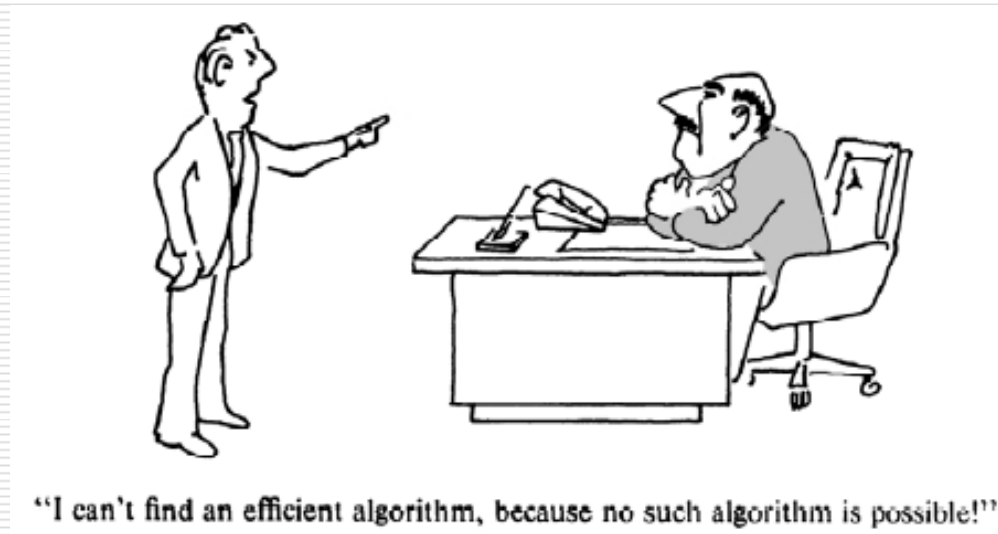
«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).



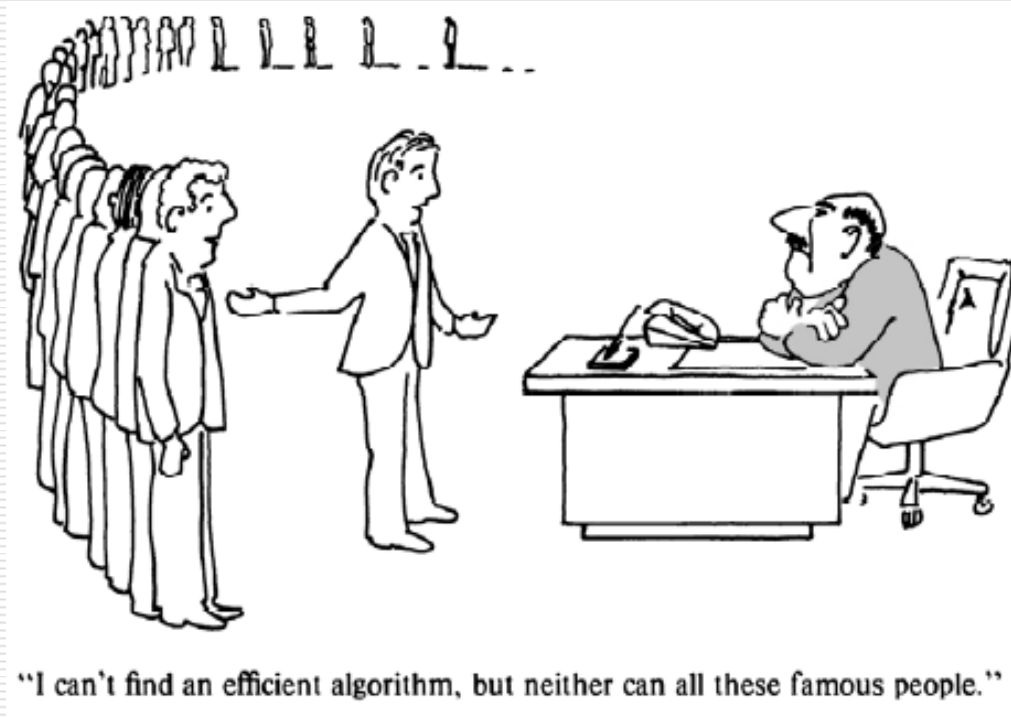
«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).



«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).



«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).
- Θεωρία **NP-πληρότητας**.
 - NP-πλήρη: κλάση εξαιρετικά σημαντικών προβλημάτων που είτε όλα επιλύονται σε πολυωνυμικό χρόνο είτε κανένα.

... και Πολυπλοκότητα

- Υπολογιστική (Χρονική) Πολυπλοκότητα:
 - Ντετερμινιστικές μηχανές Turing
 - Κλάσεις πολυπλοκότητας, **αναγωγή, πληρότητα.**
 - Χρονική Πολυπλοκότητα και **κλάση P** («εύκολα» προβλήματα).
 - Μη-ντετερμινιστικές μηχανές Turing και **κλάση NP.**
 - NP-πληρότητα («δύσκολα» προβλήματα)
 - Σχέση κλάσεων P και NP.
- Προσεγγιστικοί αλγόριθμοι για NP-δύσκολα προβλήματα.

Βιβλιογραφία

- Σ. Ζάχος. **Αλγόριθμοι και Πολυπλοκότητα**. Σημειώσεις ΕΜΠ.
- Δ. Φωτάκης. **Αλγόριθμοι και Πολυπλοκότητα**. Σημειώσεις <http://www.softlab.ntua.gr/~fotakis/data/algorithms.pdf>
- Cormen, Leiserson, Rivest, Stein. **Introduction to Algorithms**. MIT, 2001.
- Kleinberg, Tardos. **Algorithm Design**. Add.-Wesl., 2006.
- Dasgupta, Papadimitriou, Vazirani. **Algorithms**. McGraw-Hill, 2008.
- Brassard, Bratley. **Algorithmics: Theory and Practice**. Prent.-Hall, 1988.
- Papadimitriou. **Computational Complexity**. Add.-Wesl., 1994.
- Arora, Barak. **Computational Complexity: A Modern Approach**. Cambridge, 2009.
- Goldreich. **Computational Complexity: A Conceptual Perspective**. Cambridge, 2009.
- Πολλά-πολλά άλλα βιβλία και ιστοσελίδες μαθημάτων.

«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα **πρόβλημα** φαίνεται **«δύσκολο»**;
 - **«Δύσκολο»**: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).
- Πάμε στο αφεντικό και λέμε:
 - Δεν **μπορώ** να βρω αποδοτικό αλγόριθμο. **Απόλυση!**
 - Δεν **υπάρχει** αποδοτικός αλγόριθμος. **Καλό αλλά δύσκολο!**
 - **Κανένας** δεν μπορεί να βρει αποδοτικό αλγόριθμο (και όλοι πιστεύουν ότι δεν υπάρχει).
- Θεωρία **NP-πληρότητας**.
 - **NP-πλήρη**: κλάση εξαιρετικά **σημαντικών προβλημάτων** που είτε όλα επιλύονται σε πολυωνυμικό χρόνο είτε κανένα.