

Αλγόριθμοι και Πολυπλοκότητα

Δημήτρης Φωτάκης

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Οργανωτικά

- Διδάσκοντες: **Δ. Φωτάκης** και **Δ. Σούλιου**
(και **Σ. Ζάχος** στις πρόσθετες διαλέξεις για μεταπτυχιακό)
 - Βοηθοί διδασκαλίας Υ.Δ.: **Σ. Σκουλάκης, Λ. Κάβουρας, Π. Πατσιλινάκος, Ελ. Ψαρουδάκη**
 - 4 ώρες **θεωρία** (και ασκήσεις)
 - Δευτέρα 15:15 – 17:00, αμφ. 1, νέο κτ. ΣΗΜΜΥ και
 - Πέμπτη 17:15 – 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/>
- Υποβολή Ασκήσεων: www.corelab.ntua.gr/moodle/

Γραπτές Ασκήσεις

- **Ατομικές** γραπτές ασκήσεις:
 - 4 σειρές γραπτών ασκήσεων.
 - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
 - Εκφώνηση στο site. Υποβολή στο moodle.
 - 1^η γραπτή: υποβολή 2 εβδομάδες μετά την ανακοίνωση.
 - **Λύσεις:** συζήτηση στο μάθημα (και ανακοίνωση σχεδίου λύσεων)
 - Σημαντικές γιατί οδηγούν σε **βαθύτερη κατανόηση, επεκτείνουν γνώσεις και προετοιμάζουν για τις εξετάσεις!**
 - Ένα-δύο από τα θέματα των εξετάσεων παρόμοια με γραπτές (ή παλαιότερες λυμένες) ασκήσεις.
 - (Μόνο) οι «περυσινοί» μπορούν «κρατήσουν» **βαθμούς** ασκήσεων.

Προγραμματιστικές Ασκήσεις

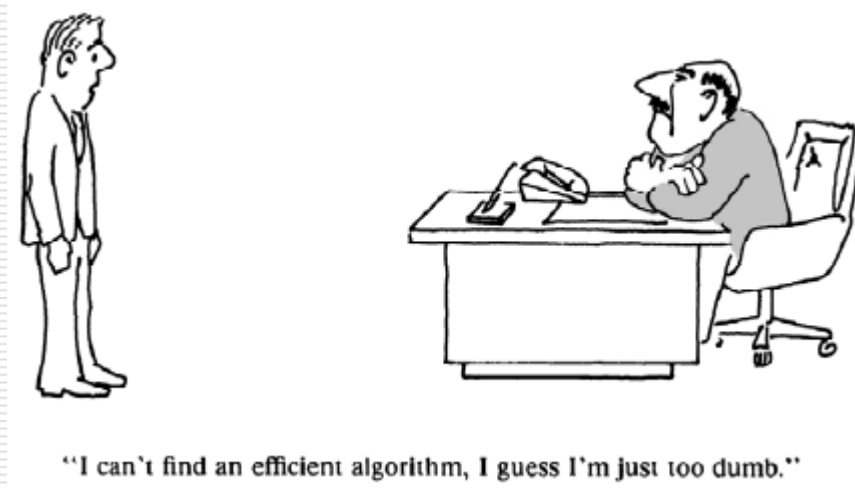
- **Ατομικές** προγραμματιστικές ασκήσεις:
 - 4 **σειρές** προγραμματιστικών ασκήσεων.
 - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
 - Εγγραφή στο **moodle**, αμέσως μόλις «ανοίξει».
Το **login** και το **password** με το οποίο θα εγγραφείτε στο moodle, θα χρησιμοποιούνται στον grader.
 - Παράδοση με **υποβολή** κώδικα στον **grader**.
 - 1^η προγραμματιστική: **δύο εβδομάδες μετά την γραπτή**
 - (Μόνο) οι «περυσινοί» μπορούν **«κρατήσουν» βαθμούς** ασκήσεων.

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 (επιλεκτικά).
 - 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

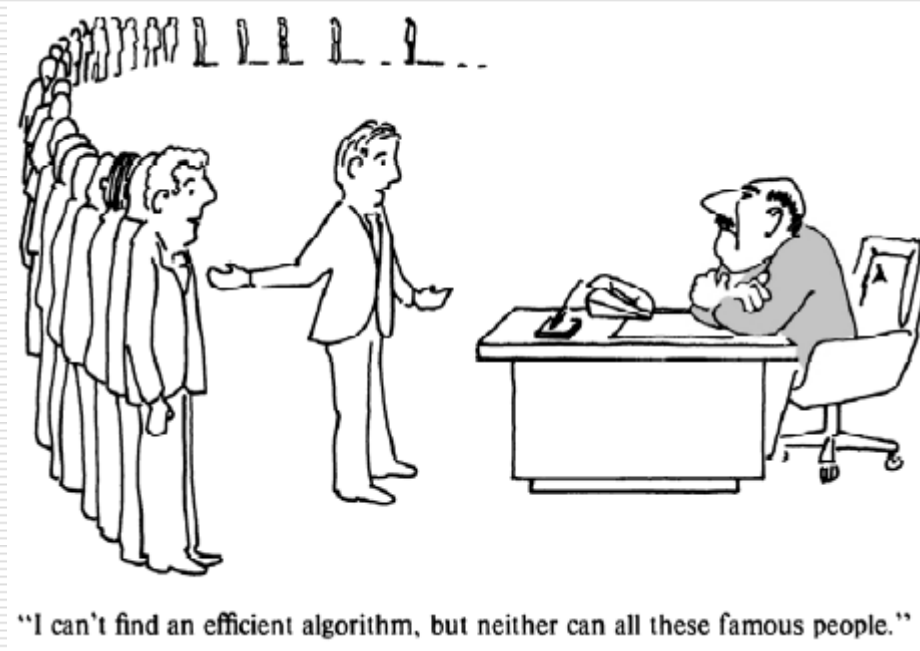
- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).



“I can't find an efficient algorithm, because no such algorithm is possible!”

«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.**
 - Πολυωνυμική αναγωγή και **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.
- ❑ J. Edmonds. **How to Think About Algorithms**. Cambridge, 2008.
- ❑ Papadimitriou. **Computational Complexity**. Add.-Wesl., 1994.
- ❑ Πολλά-πολλά άλλα βιβλία και ιστοσελίδες μαθημάτων.