

Αλγόριθμοι και Πολυπλοκότητα

Δημήτρης Φωτάκης

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Οργανωτικά

- Διδάσκοντες: **Δ. Φωτάκης** και **Δ. Σούλιου**
(και **Σ. Ζάχος** και **Θ. Λιανέας** πρόσθετες διαλέξεις για μεταπτυχιακό)
 - Δευτέρα 15:00 – 17:00 και Πέμπτη 17:00 – 19:00 (Αμφ. 1 – Webex)
 - Ιστοσελίδα: <https://helios.ntua.gr/course/view.php?id=859>
- Βαθμολογία (εφόσον μπορεί να γίνει **γραπτή εξέταση** διάρκειας $\geq 90'$):
 - **Γραπτό**: τελική εξέταση (ασκήσεις). **Βάση**: τουλάχιστον 4.5 στα 10.
 - **Ασκήσεις**: γραπτές ασκήσεις (3 σειρές)
 - **Πρόγρ**: προγραμματιστικές ασκήσεις (3 σειρές)
 - Τελικός βαθμός = $\text{Γραπτό} * 0.8 + \text{Ασκήσεις} * 0.15 + \text{Προγρ} * 0.15$
- Βαθμολογία (διαφορετικά): Pass/ Fail με βάση τελική εξέταση
 - Βαθμοί ασκήσεων θα κρατηθούν για **επόμενη γραπτή εξέταση**.

Γραπτές Ασκήσεις

- **Ατομικές** γραπτές ασκήσεις:
 - 3 σειρές γραπτών ασκήσεων.
 - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
 - Εκφώνηση στο site – υποβολή στο moodle.
 - 1^η γραπτή: Δευτέρα 25/10 → Δευτέρα 4/11.
 - **Λύσεις:** συζήτηση / παρουσίαση λύσεων στο μάθημα.
 - 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» φοιτητές μπορούν **«κρατήσουν» βαθμούς** ασκήσεων.

Προγραμματιστικές Ασκήσεις

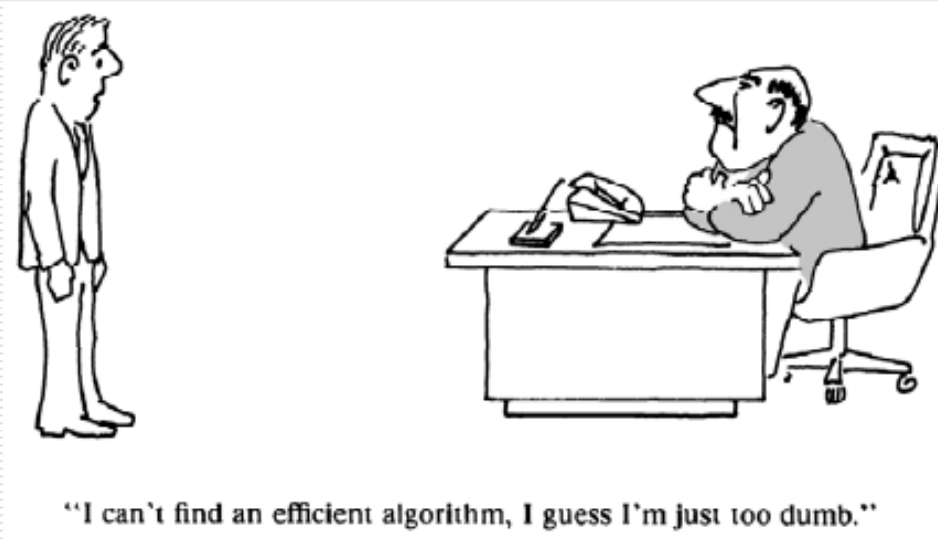
- **Ατομικές** προγραμματιστικές ασκήσεις:
 - 3 σειρές προγραμματιστικών ασκήσεων.
 - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
 - Παράδοση με **υποβολή** κώδικα στον **grader**.
 - Enroll στο **helios** και λογαριασμός **grader**:
 - Το **login** στον grader θα είναι της μορφής **el18999**.
 - Το **password** θα είναι **random**, θα το αλλάξετε την πρώτη φορά.
 - 1^η προγραμματ.: Δευτέρα 1/11 → Δευτέρα 15/11.
 - Όταν χρειάζεται, **επίδειξη** (μετά από συνεννόηση).
 - (Μόνο) οι «περυσινοί» φοιτητές μπορούν «κρατήσουν» βαθμούς ασκήσεων.

Αλγόριθμοι και ...

- Γιατί κάποια **υπολογιστικά προβλήματα** είναι **εύκολο** και κάποια άλλα είναι **δύσκολο** να λυθούν από **υπολογιστικές μηχανές**.
 - Βασικές έννοιες και βασικές δομές δεδομένων (σύντομα κ' επιλεκτικά).
 - «Εύκολα» προβλήματα: επιλύονται από **αλγόριθμους** που απαιτούν **εύλογη ποσότητα υπολογιστικών πόρων** (χρόνος, μνήμη, επικοινωνία).
 - Βασικές **τεχνικές** σχεδιασμού και ανάλυσης αλγορίθμων:
 - Διαίρει-και-Βασίλευε (εν συντομία), απληστία, δυναμικός προγραμματισμός
 - Εφαρμογή: **αλγόριθμοι γραφημάτων**
 - Ελάχιστο συνδετικό δέντρο, συντομότερα μονοπάτια, μέγιστη ροή – ελάχιστη τομή, ταιριάσματα.
 - **Πιθανοτικοί** αλγόριθμοι

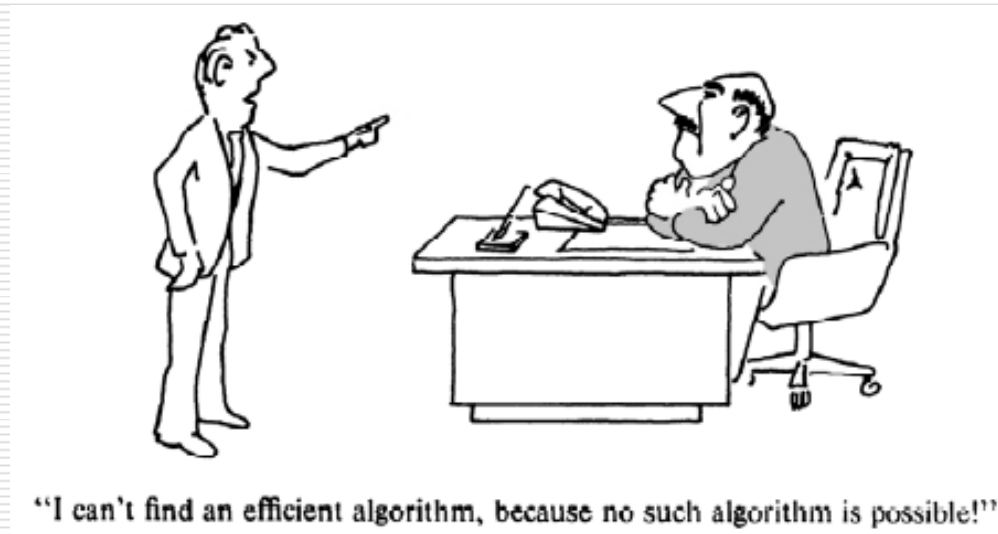
«Δύσκολα» Προβλήματα

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



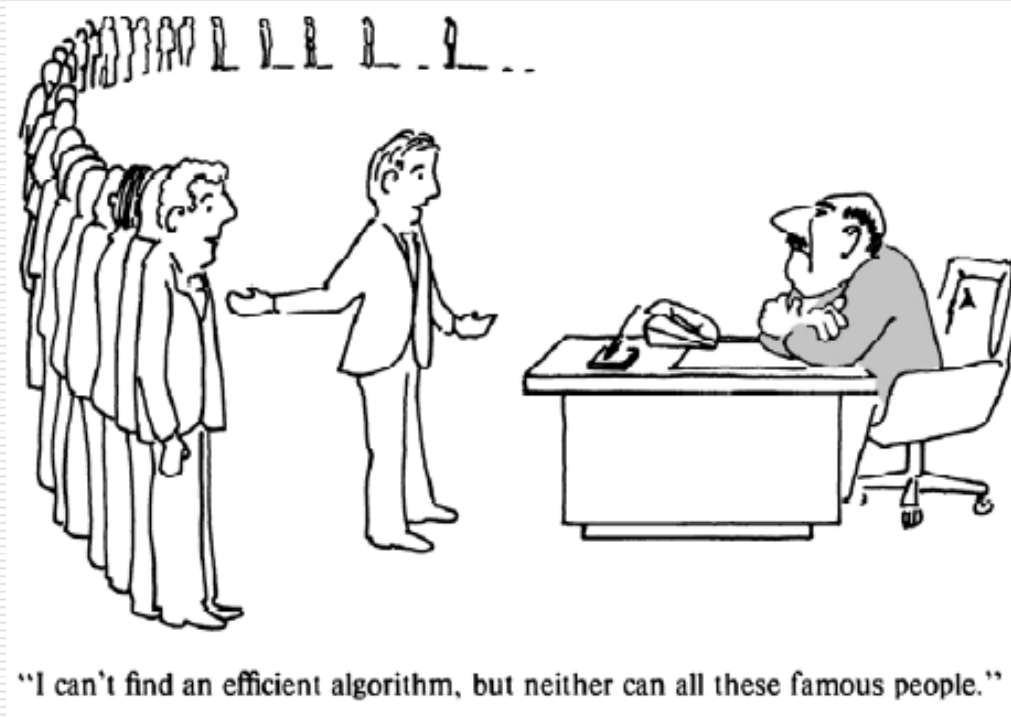
«Δύσκολα» Προβλήματα

- 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, μη υπολογισιμότητα (απλή αναφορά – έμφαση στην έννοια της αναγωγής και της πληρότητας)
 - 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.
- J. Erickson. **Algorithms**. <http://jeffe.cs.illinois.edu/teaching/algorithms/>
- Papadimitriou. **Computational Complexity**. Add.-Wesl., 1994.
- Πολλά-πολλά άλλα βιβλία και ιστοσελίδες μαθημάτων.