

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)
 - Βοηθοί διδασκαλίας Υ.Δ.:
Λ. Κάβουρας, Α. Καλαβάσης, Α. Μαθιουδάκη, Π. Πατσιλινάκος, Α. Ρισσάκη, Ε. Ψαρουδάκη
 - Δευτέρα 15:00 – 17:00 και Πέμπτη 17:00 – 19:00 (WebEx)
 - Ιστοσελίδα: <https://courses.corelab.ntua.gr/algorithms>
- Βαθμολογία (εφόσον μπορεί να γίνει **γραπτή εξέταση** διάρκειας $\geq 90'$):
 - **Γραπτό**: τελική εξέταση (ασκήσεις). Τουλάχιστον 4.3 στα 10.
 - **Ασκήσεις**: γραπτές ασκήσεις (4 σειρές)
 - **Πρόγρ**: προγραμματιστικές ασκήσεις (4 σειρές)
 - Τελικός βαθμός = $\text{Γραπτό} * (0.75 + \text{Ασκήσεις} * 0.02 + \text{Προγρ} * 0.02)$
- Βαθμολογία (διαφορετικά): Pass/ Fail με βάση τελική εξέταση
 - Βαθμοί ασκήσεων θα κρατηθούν για **επόμενη γραπτή εξέταση**.

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: Δευτέρα 19/10 → Δευτέρα 2/11.
 - **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

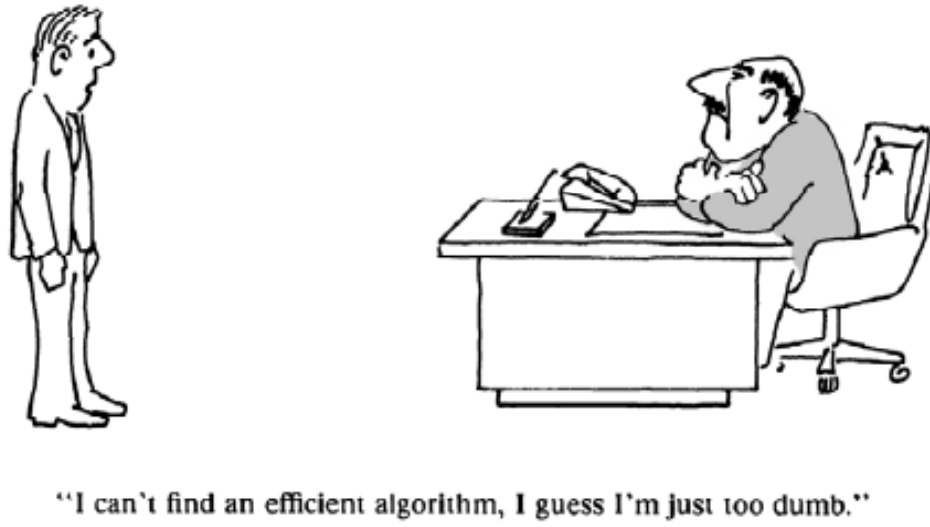
- **Ατομικές** προγραμματιστικές ασκήσεις:
 - 4 σειρές προγραμματιστικών ασκήσεων.
 - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
 - Εγγραφή στο **moodle**, αμέσως μόλις «ανοίξει».
Το **login** και το **password** με το οποίο θα εγγραφείτε στο moodle, θα χρησιμοποιηθούν για λογαριασμό στον grader.
 - Παράδοση με **υποβολή** κώδικα στον **grader**.
 - 1^η προγραμματ.: Πέμπτη 29/10 → Πέμπτη 12/11.
 - Όταν χρειάζεται, **επίδειξη** (μετά από συνεννόηση).
 - (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 (σύντομα κ' επιλεκτικά).
 - 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

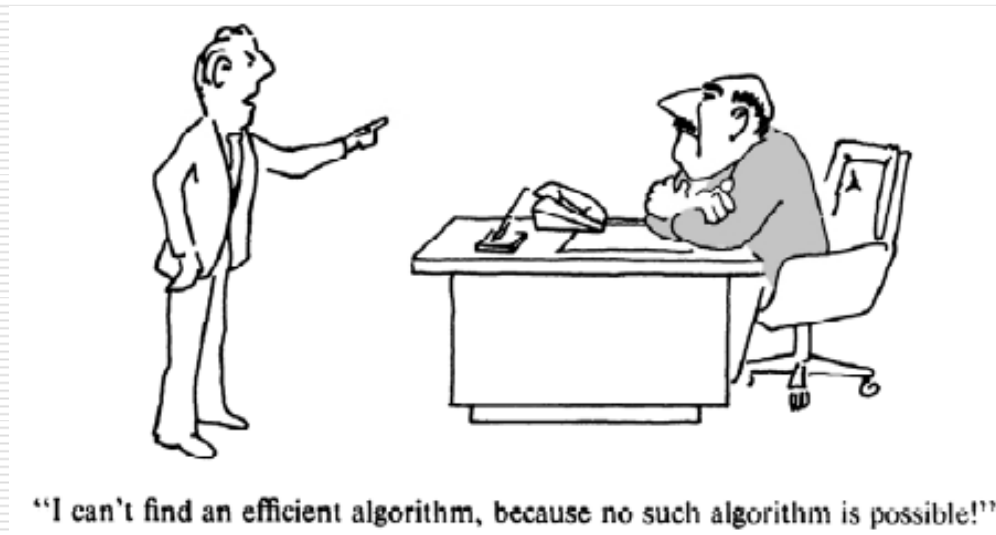
«Δύσκολα» Προβλήματα

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



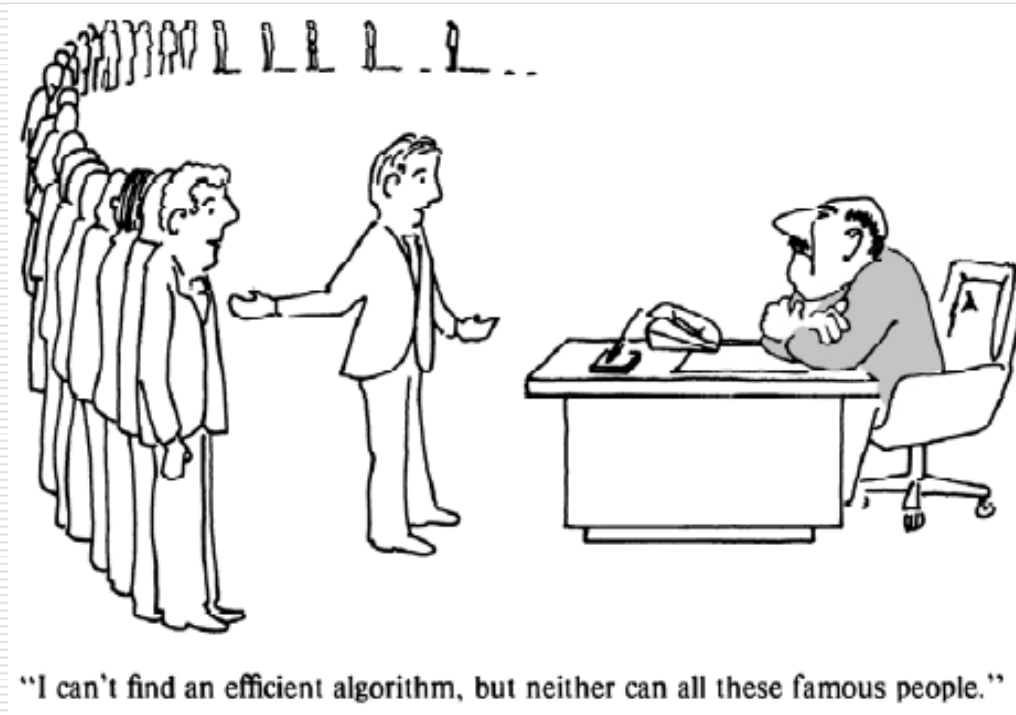
«Δύσκολα» Προβλήματα

- 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
 - 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.