Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: **Δ. Φωτάκης, Δ. Σούλιου**

Επιμέλεια διαφανειών: **Δ. Φωτάκης**

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο

Οργανωτικά

- Διδάσκοντες: **Δ. Φωτάκης, Δ. Σούλιου**
  (και **Σ. Ζάχος, Θ. Λιανέας** πρόσθετες διαλέξεις για μεταπτυχιακό)
  - Δευτέρα 15:00 – 17:00 και Πέμπτη 17:00 – 19:00 (Αμφ. 1)
  - Ασκήσεις / επιπλέον διαλέξεις / αναπληρώσεις: Τρίτη 15:30 – 17:30
  - Ιστοσελίδα: https://helios.ntua.gr/course/view.php?id=859
- Βαθμολογία:
  - Γραπτό: τελική εξέταση (ασκήσεις). **Βάση: τουλάχιστον 40%.**
  - Γραπτές: γραπτές ασκήσεις (3 σειρές).
  - Πρόγρ: προγραμματιστικές ασκήσεις (3 σειρές).
  - Βαθμολογία ασκήσεων με **σύντομη παρουσίαση λύσεων** και συζήτηση.
  - Βαθμός = Γραπτό*0.80+Γραπτές*0.15+Προγρ*0.15, αν **γραπτό >= 40%.**
  - Βαθμός = Γραπτό, αν **γραπτό < 40%.**
  - Βαθμοί ασκήσεων διατηρούνται και για **αμέσως επόμενο** ακαδ. έτος.

Γραπτές Ασκήσεις

- **Ατομικές** γραπτές ασκήσεις:
  - 3 σειρές γραπτών ασκήσεων.
  - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
  - Εκφώνηση στο site – υποβολή στο helios.
  - 1η γραπτή: Δευτέρα 20/10 → Πέμπτη 6/11.
  - Λύσεις: συζήτηση 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.

Προγραμματιστικές Ασκήσεις

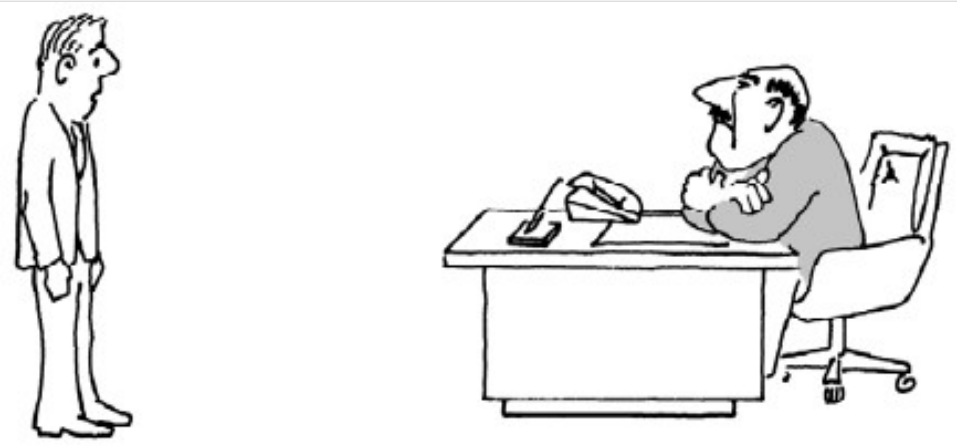
- **Ατομικές** προγραμματιστικές ασκήσεις:
  - 3 σειρές προγραμματιστικών ασκήσεων.
  - Περίπου 2 εβδομάδες για κάθε σειρά ασκήσεων.
  - Παράδοση με **υποβολή** κώδικα στον **grader**.
  - **Εγγραφή** στο **helios** → λογαριασμός **grader**:
    - Παράκληση: **η εγγραφή** στο helios πριν τις **20 Οκτώβρη**.
    - Το **login** στον grader θα είναι της μορφής **el22999**.
    - Το **password** θα είναι **random**, θα το αλλάξετε την πρώτη φορά.
  - 1η προγραμματ.: Δευτέρα 27/10 → Πέμπτη 13/11.

Αλγόριθμοι και ...

- Γιατί κάποια υπολογιστικά προβλήματα είναι **εύκολο** και κάποια άλλα είναι **δύσκολο** να λυθούν από υπολογιστικές μηχανές.
  - Βασικοί αλγόριθμοι (BFS, DFS, εφαρμογές τους, αλγόριθμοι MST) και βασικές δομές δεδομένων (ουρές προτεραιότητας, union find, λεξικό) **θεωρούνται γνωστά!**
  - Βασικές αρχές δομών δεδομένων, σύντομη επανάληψη, επιπλέον παραδείγματα.
  - **«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
    - Συντομότερα μονοπάτια, μέγιστη ροή – ελάχιστη τομή, ταιριάσματα, ροή ελάχιστου κόστους.
  - Σύντομη εισαγωγή σε **πιθανοτικούς** αλγόριθμους.

«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, I guess I'm just too dumb."

«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).

"I can't find an efficient algorithm, but neither can all these famous people."

«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:
  - 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.
  - Άλλες κλάσεις πολυπλοκότητας (space complexity, counting, total functions)
- **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.