Algoritmické riešenie ťažkých problémov, 2-AIN-205

Algorithmics for Hard Problems

Lecturers:

Tomáš Vinař, M-163, tomas.vinar@fmph.uniba.sk

Ján Pastorek, I-5, jan.pastorek@fmph.uniba.sk

Lectures: Tuesday, 14:50-16:20 (F1-247)

Tutorials: Thursday, 8:10-9:40 (M-XII)

Class communication platform: vektor.fmph.uniba.sk

Use code **j2fr5uc** to sign up

Web: https://compbio.fmph.uniba.sk/vyuka/artp/

Textbooks:

- Cormen, Leiserson, Rivest, Stein: Introduction to Algorithms, MIT Press 2009
- Vazirani: Approximation Algorithms, Springer 2001
- Motwani, Raghavan: Randomized Algorithms, Cambridge University Press 2005

Other sources:

- Class webpage
- Use vektor for discussion
- Try to **actively** solve problems:
  tutorials, homework assignments,
  other problems at the end of the recommended chapters in textbooks

Class topics

Advanced methods for solving algorithmic problems:

- approximation algorithms
- randomized algorithms
- integer linear programming
- parametric complexity

What we assume you already know:

- greedy algorithms (including how to prove correctness)
- dynamic programming
- basis of computational complexity (NP-hard problems)

covered in "Efficient algorithms and data structures" (EADŠ/EAZ)
or "Design of efficient algorithms" (TEA)

Grades

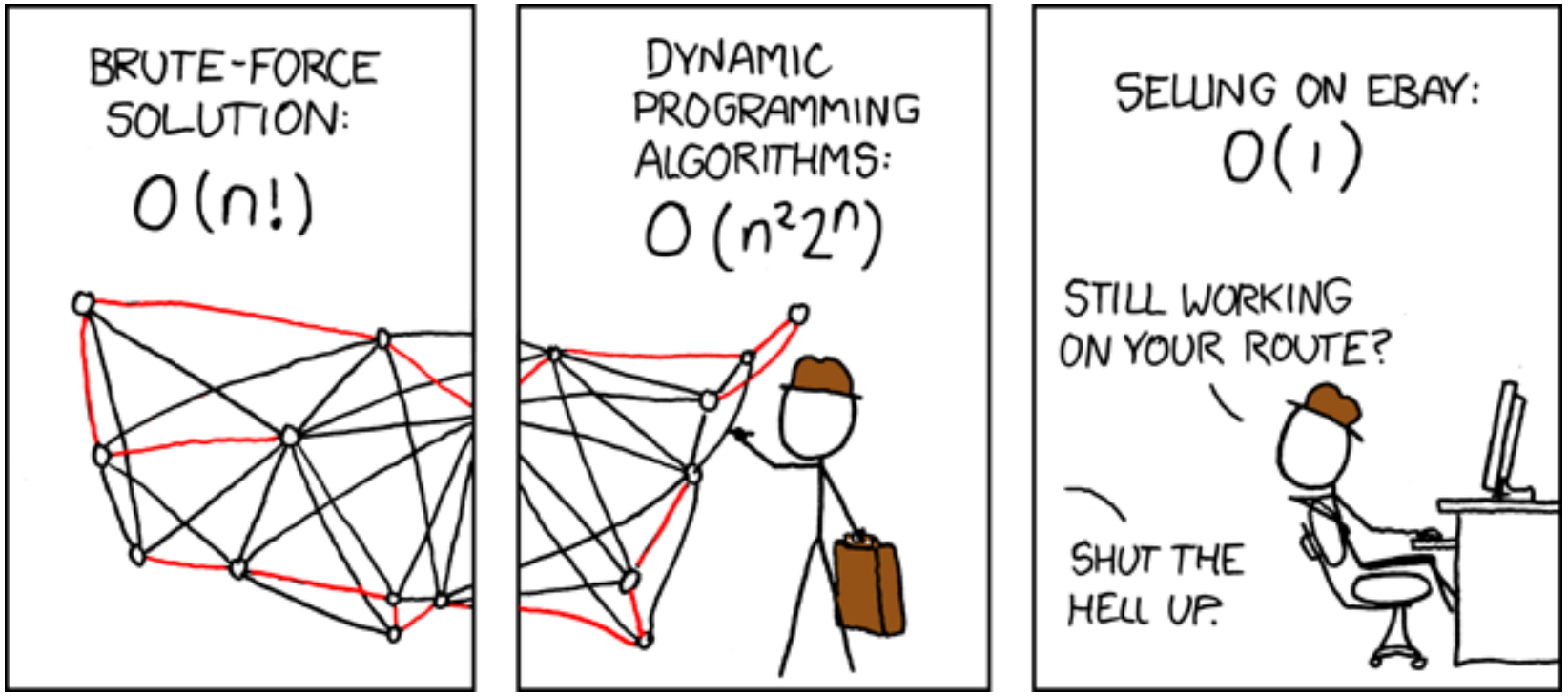
- 40%: Homework assignments (4x)
  (includes one programming task each)
- 10%: Electronic quizzes (evening before tutorials)
- 50%: Final exam
- to pass, you must have at least 50% points on the final
- 90+ = A, 80+ = B, 70+ = C, 60+ = D, 50+ = E

Cheating

- Penalty: -100% + disciplinary commitee
- We **support discussion** about homework, **but:**
  - Don't keep any notes
  - Wait for several hours before writing your own solution
  - Write the names of colleagues you discussed the task with
  - Treat **ChatGPT** the same way as a colleague

Running time to solve the problem of size n

	Sol.4 $O(n)$	**Sol.3** $O(n \log n)$	**Sol.2** $O(n^2)$	**Sol.1** $O(n^3)$	**Sol.0** $O(2^n)$
10	ε	ε	ε	ε	ε
50	ε	ε	ε	ε	2 weeks
100	ε	ε	ε	ε	2800 univ.
1000	ε	ε	0.02s	4.5s	—
10000	ε	0.01s	2.1s	75m	—
100000	0.04s	0.12s	3.5m	52d	—
1 mil.	0.42s	1.4s	5.8h	142yr	—
10 mil.	4.2s	16.1s	24.3d	140000yr	—



http://xkcd.com/399

Efficient algorithms

You already know: greedy algorithms, divide and conquer, dynamic programming, data structures, graph algorithms

This semester: randomization, approximation algorithms, ILP

Analysis of algorithms

You already know: worst-case time complexity, asymptotic notation, basic complexity classe P, NP

This semester: average-case time complexity, expected running time, analysis with approximation factors, more detailed hierarchy of complexity classes

Lower bounds

You already know: NP-hardness, non-computability

This semester: inapproximability