

# Algorithms and Data Structures II

Fall semester 2023-2024. Tuesday 09:00-10:30 at D1.13

**Instructor:** Prof. Evanthia Papadopoulou

Email: [evanthia.papadopoulou@usi.ch](mailto:evanthia.papadopoulou@usi.ch)

**Co-instructor:** Dr. Elena Arseneva

Email: [elena.arseneva@usi.ch](mailto:elena.arseneva@usi.ch)

Office Hours: Monday 17:00 - 18:00 by appointment

**Teaching Assistant:** Alex Bortolotti

Email: [alex.bortolotti@usi.ch](mailto:alex.bortolotti@usi.ch)

Office Hours: Monday 16:00 - 18:00

## Textbook:

*Introduction to Algorithms*, 4th edition, by T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, MIT Press, 2022.

**Other readings:** *Algorithm Design*, by J. Kleinberg, E. Tardos, Addison Wesley, 2013.

**Prerequisites:** Algorithms and Data Structures

**Grading:** homeworks and in-class quizzes 15-20%, midterm: 20%, final exam 65-60%

Optional project: up to 1 bonus point to total grade.

To pass the course a 50% on the final exam is required.

In-class quizzes include questions on a homework set or reading assignment. This is instead of submitting written answers to homework problems. Solutions to the homework problems will be provided in class.

Project is either theoretical (understanding in depth an additional advanced topic) or practical (e.g., developing a demo of a certain algorithm or data structure), with a presentation at the end of the course. Projects can be done in groups of two.

We reserve the right to slightly adjust your grade up or down based on participation in class and our observations of your performance. For example, a strong upwards trend may give you a boost; doing well on the homework but poorly on the midterm and final may cause your grade to be reduced. We also reserve the right to make slight adjustments to the distribution of grades depending on how the course develops.

## Contents (Tentative):

- Disjoint set data structures (union-find, amortized analysis)
- Augmenting data structures (dynamic order statistics, interval trees)
- Computing convex hulls (Graham scan, Quick hull, etc)
- Dynamic programming. Graph algorithms related to dynamic programming such as the the Floyd-Warshall shortest path algorithm.
- Max-flow min-cut and the Ford-Fulkerson method, bipartite graphs and matching
- Online algorithms
- Clustering and nearest-neighbor search (if time permits)
- Brief introduction to Intractability and NP completeness.

**Academic honesty:** The work that you turn in must be your own. If any outside source is used (another student, a book, the web, etc.), you must note this on your assignment. If you find a solution on the web, please cite it and use your own words to describe it. As long as you correctly cite resources this is perfectly acceptable. Plagiarism will not be tolerated, and will result in a failing grade on the assignment or exam. I strongly discourage you from relying too much on your classmates or the web for help. If you do, you will almost certainly struggle on the midterm and final exams.