

Algorithms & Data Structures

Course Presentation

Marc Gaetano

Edition 2017-2018

Welcome!

In this course we will learn *fundamental data structures and algorithms for organizing and processing information*

- “Classic” data structures / algorithms
- How to rigorously analyze their efficiency
- How to decide when to use them
- Stacks, Queues, trees, sorting, etc.

Goals

- Understand **basic algorithms** and **data structures**
- Understand/Analyze their **complexity**
- Be able to code them in a **programming language (Python)**
- Be able to **make good design choices** as a developer, project manager, etc.
- Be able to **justify** and **communicate** your design decisions

You will learn the key abstractions used almost every day in just about anything related to computing and software.

This is not a course about Python! We use Python as a tool, but the data structures you learn about can be implemented in any language.

Course organization

- **Course web page**

`http://users.polytech.unice.fr/~gaetano/ads`

login: si3 password: ADS1718

- **Course duration**

- 13 weeks

- **Lectures/Labs**

- 4 hours every Wednesday afternoon

- You should review the slides before coming to class

- Exercises to be done during the lab and to complete by yourself

Staff

- **Marc Gaetano**
gaetano@polytech.unice.fr

– G1 & G2
- **Christophe Papazian**
papazian@polytech.unice.fr

– G3 & G4

Course materials

- We'll be using Python for the programming assignments
 - Python 3.x
 - IDLE (or any Python IDE)

- Textbook:

Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press, July 2009

A PDF copy is available on the course web page

- A good Python reference of your choosing

Don't struggle **Googling** for features you don't understand, just link to a reference page you like and stick to it!

Grading system

- **3 exams**
 - At about 1/3, 2/3 and end of the course
 - Paper-based
 - Closed everything
- **Final grade**
 - Average of the 3 exams

What this course will cover

- Introduction to Algorithm Analysis
- Lists, Stacks, Queues
- Binary Search Trees
- Heaps, Priority Queues
- Sorting
- Disjoint Sets
- Graphs implementations
- Graphs algorithms