Algorithmic Design

Alberto Casagrande
Email: `acasagrande@units.it`

a.y. 2020/2021

Alberto Casagrande

E-mail: `acasagrande@units.it`
Office: Building H2-bis, 3rd floor, Room 330
Phone: 040 558 2620

Program "efficiency"

- abstract the notion of program
- define a measure of efficiency/complexity
- show techniques to compute this measure
- present some widespread problems and some solutions

Why learning algorithmic design?

Decent computer programmers stand on two legs:

- software engineering
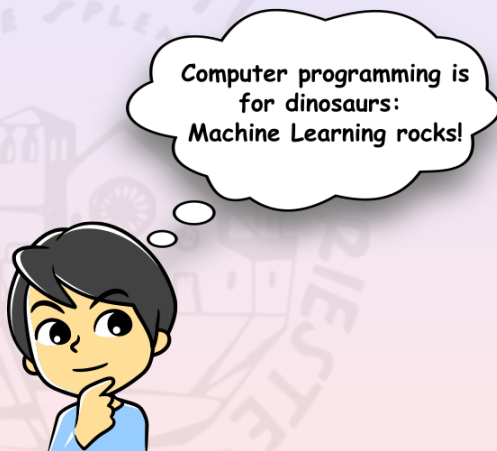- algorithmic design

Why learning algorithmic design?

Decent computer programmers stand on two legs:

- software engineering
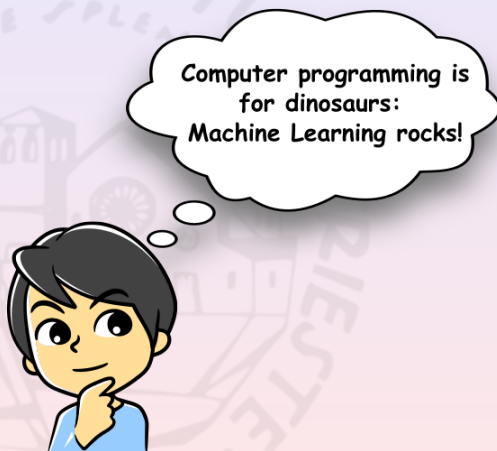- algorithmic design

Algorithmics tells us:

- whether a program will be time-effective before coding it
- how to estimate the execution time of a program
- whether the program strategy can be improved

Ya,. . .

Ya,... not really always

Nostalgia and Sadness Hit Back

- ML models usually deal fixed-sized problem instances
- ML tools are implemented by old-school programming
- how to generate/extract ML dataset?
- ML models return probabilistic/statistical results
- now feasible in some cases (e.g., which training set to resort a card deck?)

- Lessons
- Exercises
- Coding

Usually, from 9.15 to 11 (no, late admissions!), on

- Monday
- Wednesday
- Thursday

See a complete schedule of all the courses at

<div align="center">https://dssc.units.it/lecture-plan.</div>

What kind of course is this?

You have really different backgrounds

- mathematics
- physics
- biological studies
- engineering
- economic studies
- . . .

Many of you need an introduction to the topic. . .

What kind of course is this?

You have really different backgrounds

- mathematics
- physics
- biological studies
- engineering
- economic studies
- . . .

Many of you need an introduction to the topic. . .

The course consists in two (not disjoint) parts. . .

Topics

Preliminaries

- Asymptotic complexity
- Matrix multiplication
- Sorting algorithms
- Binary Search Trees and Red Black Trees
- Graph Browsing and Strongly Connected Components
- Shortest Path Problems
- Transitive Closure of a Graph
- The Routing Problem
- String Matching
- Suffix Trees and Suffix Arrays

Less "usual" topics

- Asymptotic complexity
- Matrix multiplication
- Sorting algorithms
- Binary Search Trees and Red Black Trees
- Graph Browsing and Strongly Connected Components
- Shortest Path Problems
- Transitive Closure of a Graph
- The Routing Problem
- String Matching
- Suffix Trees and Suffix Arrays

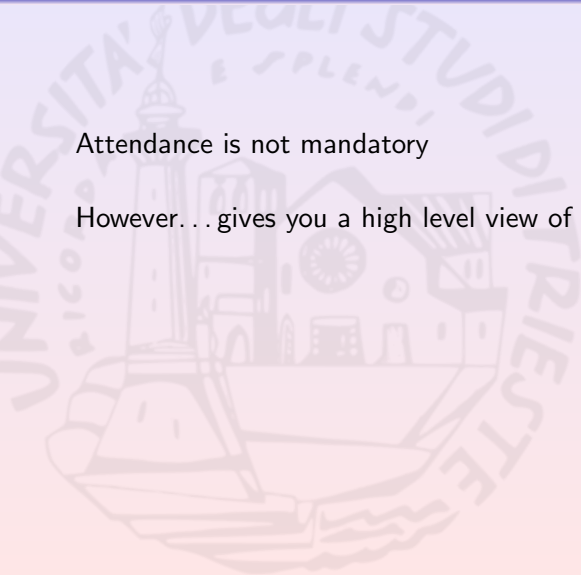# Must I attend the course?
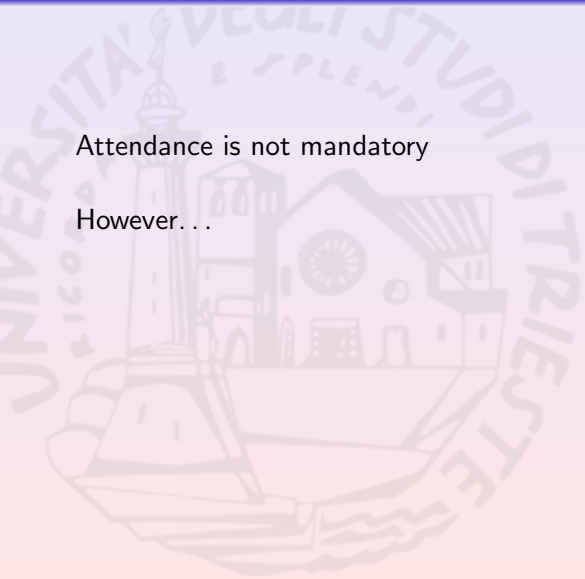
Attendance is not mandatory

However. . .

Must I attend the course?

Attendance is not mandatory

However. . . gives you a high level view of the topics for **free!!!**

Must I attend the course?

Attendance is not mandatory

However... gives you a high level view of the topics for **free!!!**

If you are not a "novice", you do not need Preliminaries, but . . .

Must I attend the course?
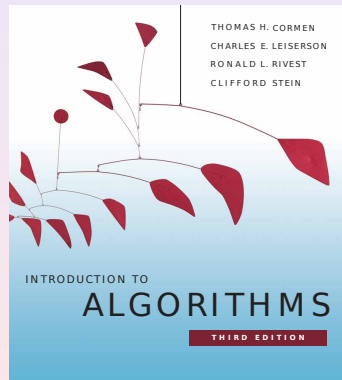
Attendance is not mandatory

However. . . gives you a high level view of the topics for **free!!!**

If you are not a "novice", you do not need Preliminaries, but . . .

it may help in "unusual" topics.

Introduction to Algorithms, (3rd Edition)
Cormen, Leiserson, Rivest, Stein
MIT Press

Course material

You can find it on Moodle as soon as released

    https://moodle2.units.it/course/view.php?id=7190

Search for "*587SM-2 - ALGORITHMIC DESIGN 2020*".

A final oral exam

+

Homework (Optional)

Will be handed out during the course in batches.

At most 2 weeks to solve and return each batch.

If every batch is completed, you will earn 3 extra point (over 30) for the final exam.

Final Exam Dates

- June 14, 2021, 9AM
- July 5, 2021, 9AM
- July 26, 2021, 9AM
- September 1, 2021, 9AM
- September 15, 2021, 9AM
- January 21, 2022, 9AM

Registration is mandatory.

Any question?