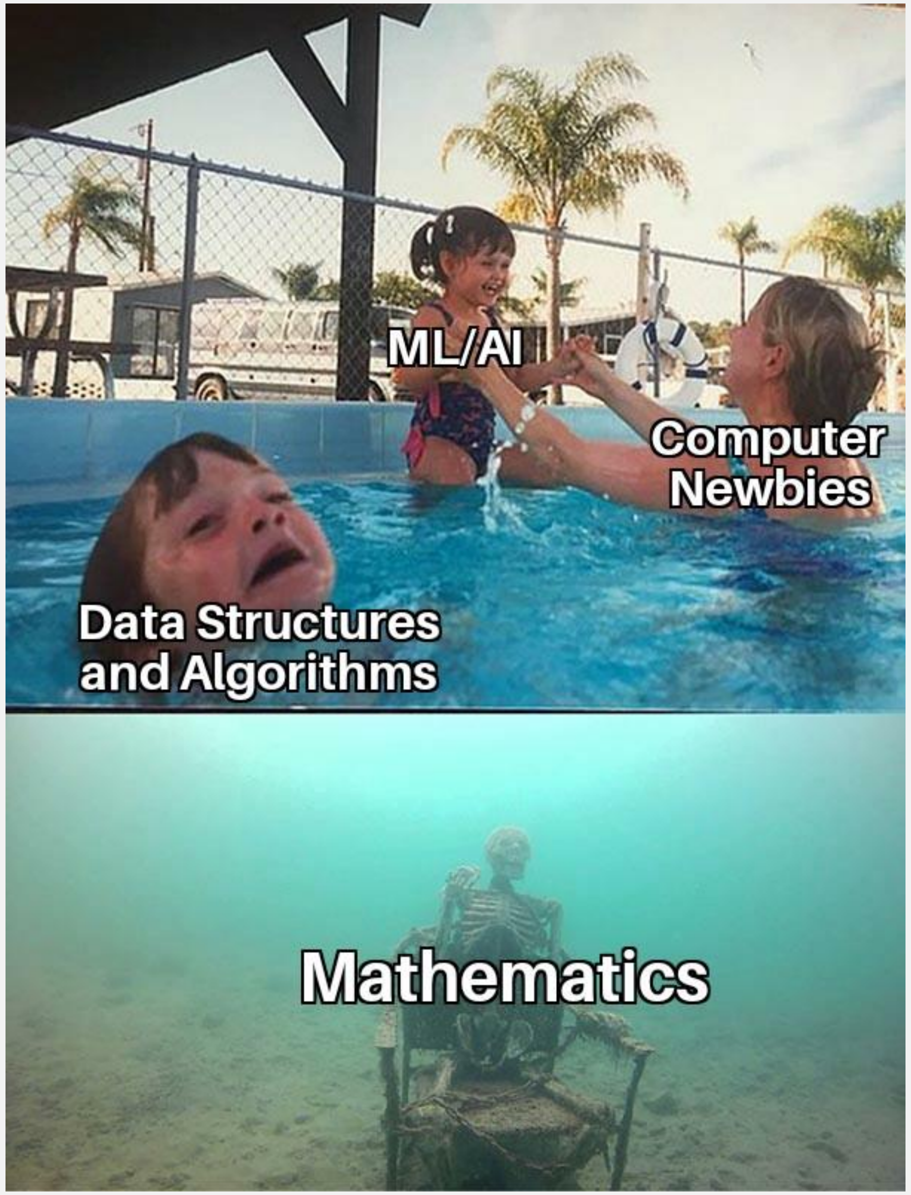
ALGORITHMS & DATA STRUCTURES

Prof. Antonio Momblán

ABOUT MYSELF

- Hi, I'm Antonio Momblán
- Born in Ávila, Spain
- Masters Degree in Computer Science at Universidad Pontificia de Salamanca and Linnaeus University at Växjö, Sweden
- 19 years of experience as a Software Engineer in the private sector
- Currently I'm a Backend Engineer at www.adyen.com

ABOUT THE COURSE



EVALUATION CRITERIA

Criteria	Score %	Comments
Other	30 %	Practice sessions deliveries
Workgroups	30 %	Board game
Final Exam	30 %	Grading test
Other	10 %	Participation in class

ABOUT THE COURSE

- Divided in 7 modules:

1. Introduction (Sessions 1-2)
2. Searching and sorting (Sessions 3-11)
3. Linear data structures (Sessions 3-11)
4. Unordered data structures (Sessions 13-15)
5. Tree-like data structures (Sessions 16-20)
6. Graphs (Sessions 21-23)
7. Dynamic Programming and Greedy Algorithms (Sessions 25-26)

Practical Tests:

- Linear Data Structures and Sorting: Session 14
- Trees and Graphs: Session 24
- Dynamic Programming and Greedy: Session 29

Final Exam: Session 30

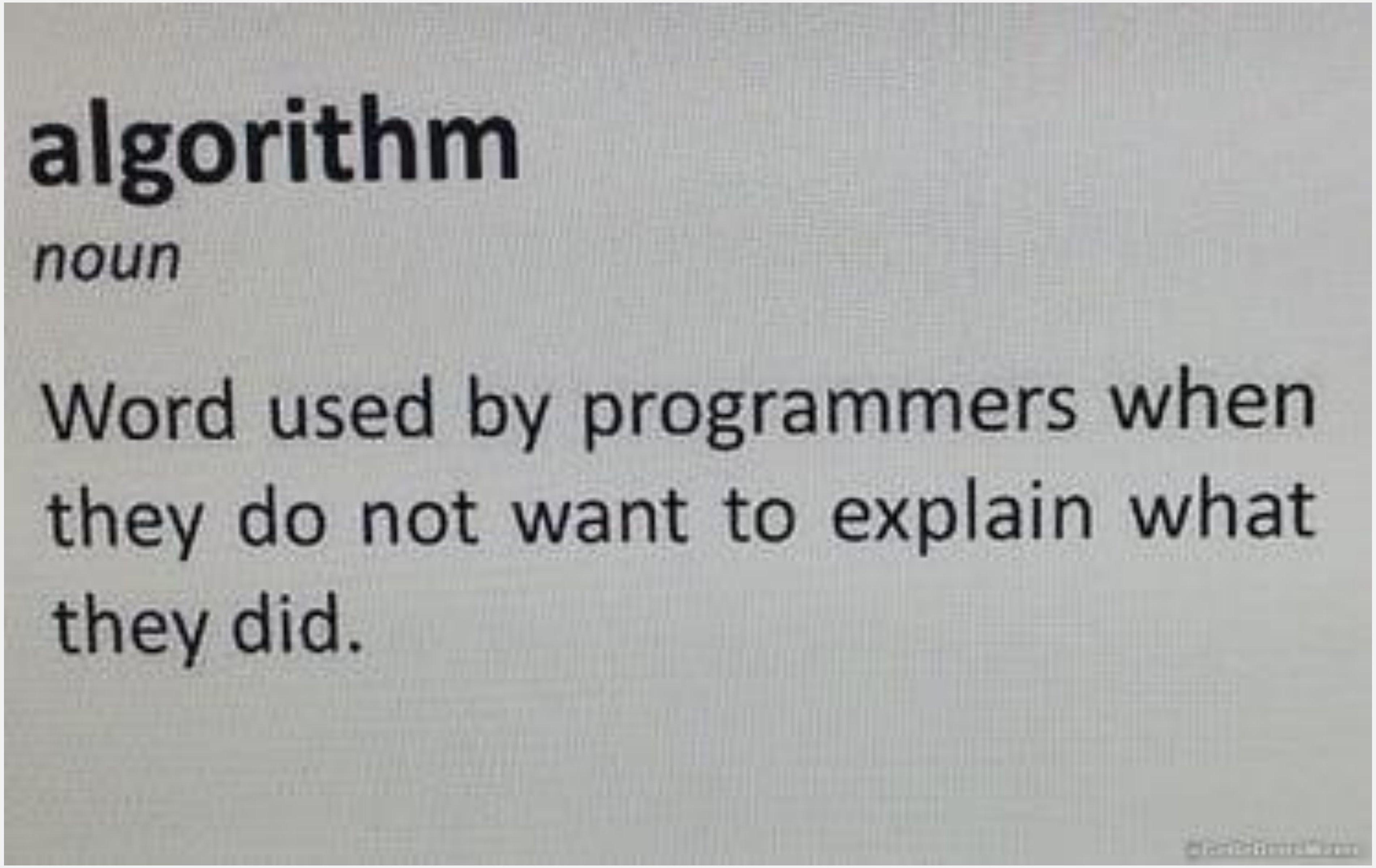
TOOLS YOU NEED

- Bring your laptop, if possible.
- Coding IDE of your choice (my recommendations: PyCharm, Vim or VSCode)
- HackerRank account (FREE) (https://www.hackerrank.com/)
- LeetCode account (FREE) (https://leetcode.com/)

BASIC RULES

- During Class:
  - No smartphones
  - No applications/webs that do not serve the purposes of the course
- GenAI policy:
  - Allowed to explain/summarize during learning.
  - Disallowed to produce code.
  - Totally prohibited during exams/evaluations.

WHAT IS AN ALGORITHM?



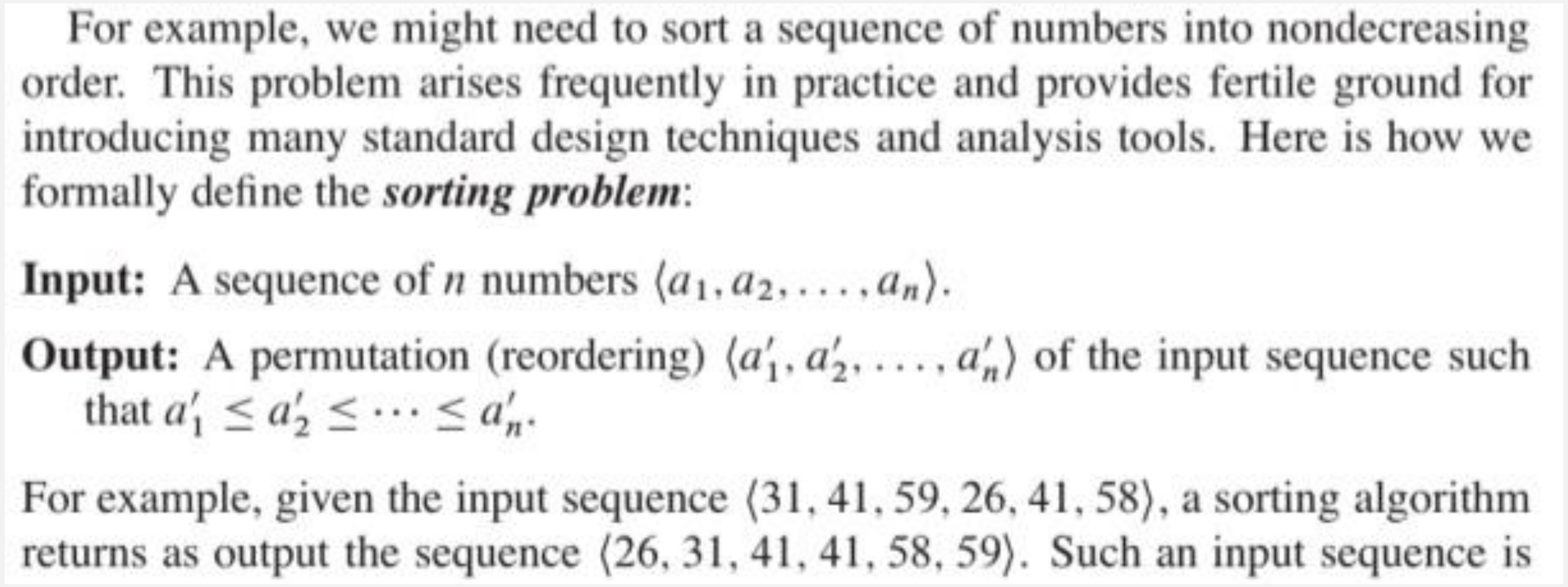
algorithm

noun

Word used by programmers when they do not want to explain what they did.

FORMALLY SPEAKING…

"An algorithm is a function that takes an input or set of inputs and produces an output or set of outputs"



For example, we might need to sort a sequence of numbers into nondecreasing order. This problem arises frequently in practice and provides fertile ground for introducing many standard design techniques and analysis tools. Here is how we formally define the ***sorting problem***:

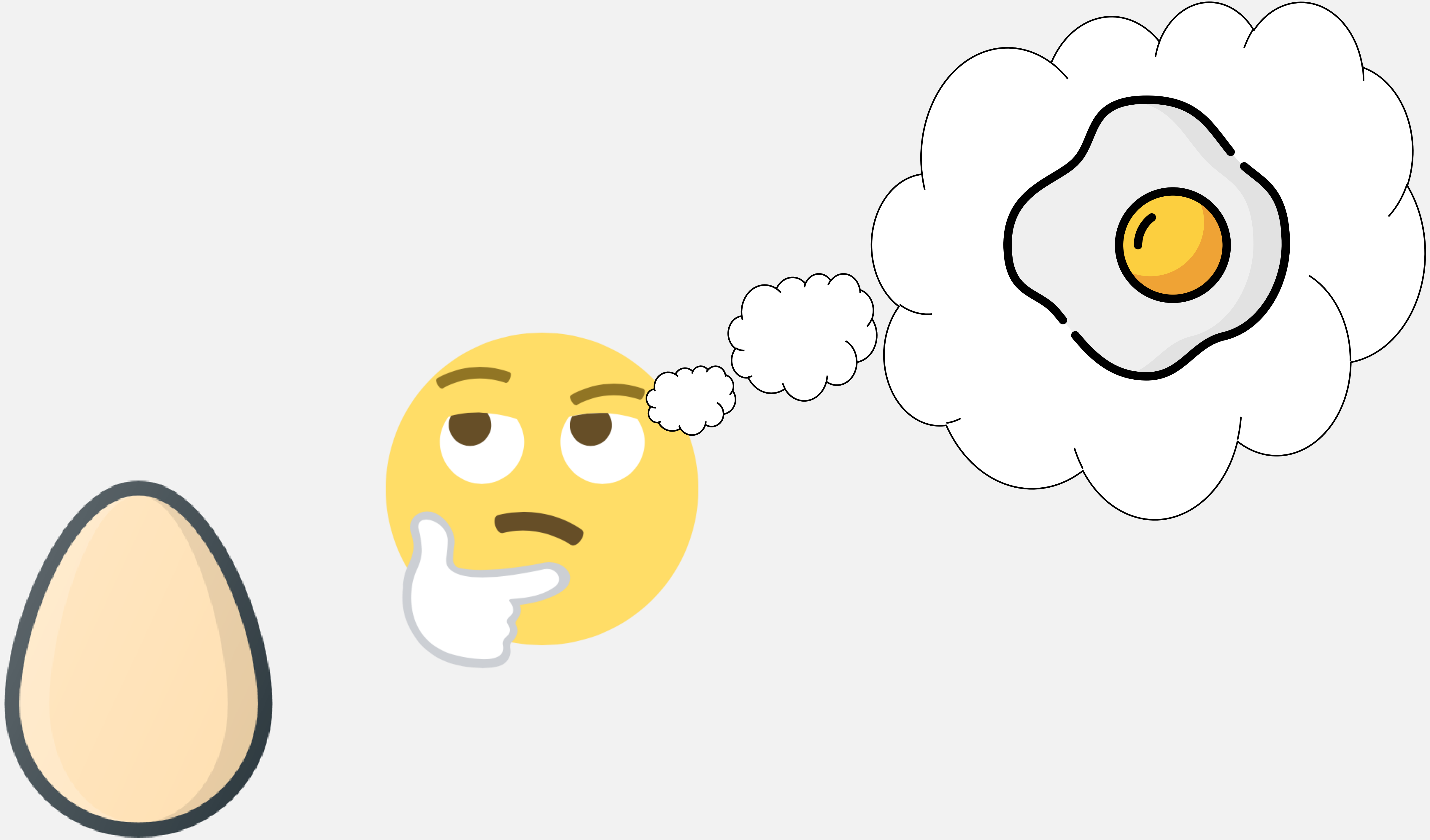
Input: A sequence of n numbers $\langle a_1, a_2, \ldots, a_n \rangle$.

Output: A permutation (reordering) $\langle a'_1, a'_2, \ldots, a'_n \rangle$ of the input sequence such that $a'_1 \le a'_2 \le \cdots \le a'_n$.

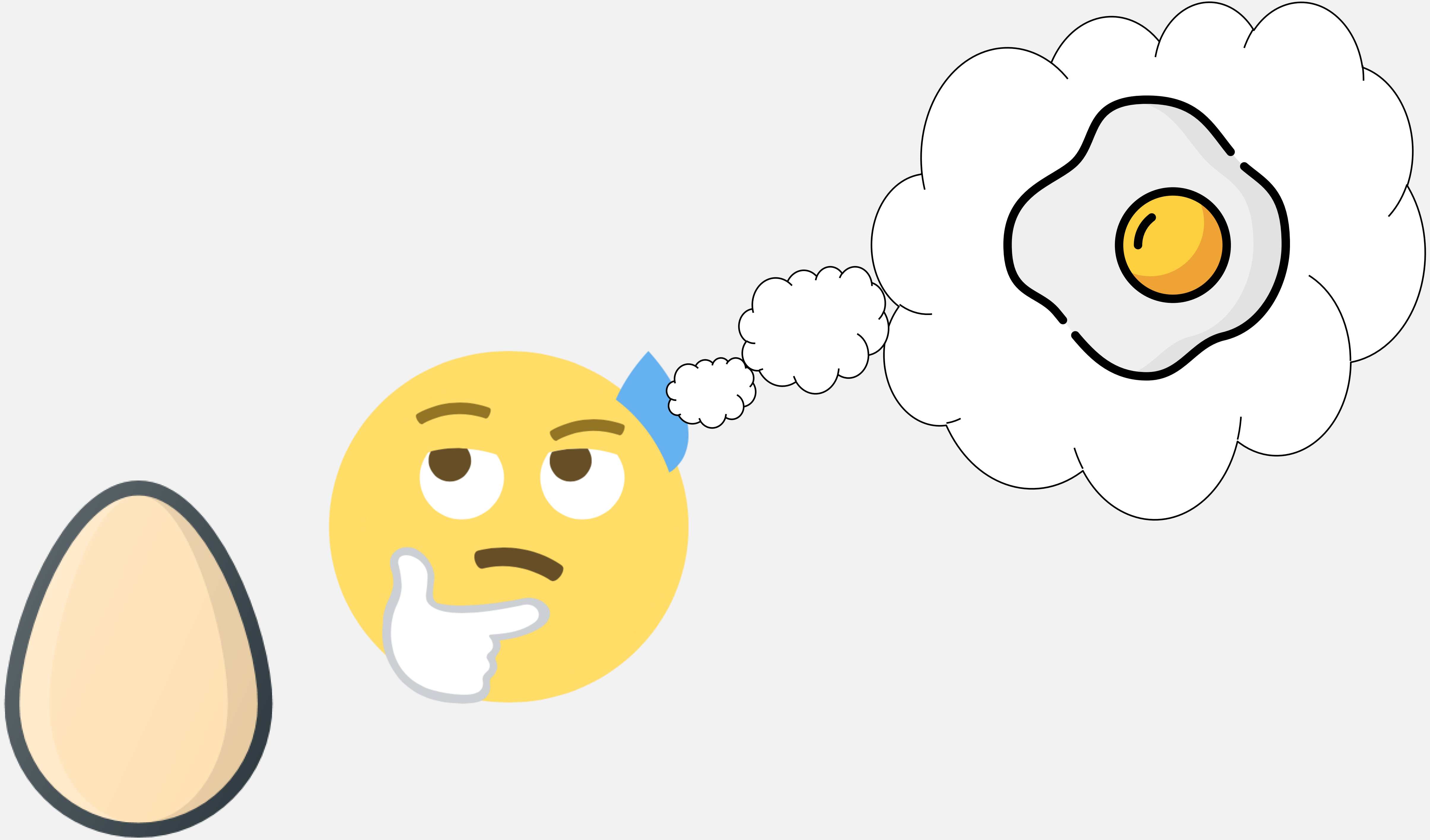
For example, given the input sequence $\langle 31, 41, 59, 26, 41, 58 \rangle$, a sorting algorithm returns as output the sequence $\langle 26, 31, 41, 41, 58, 59 \rangle$. Such an input sequence is

Cormen et al.

https://studio.code.org/levels/8729 https://studio.code.org/levels/8726



INSTRUCTIONS



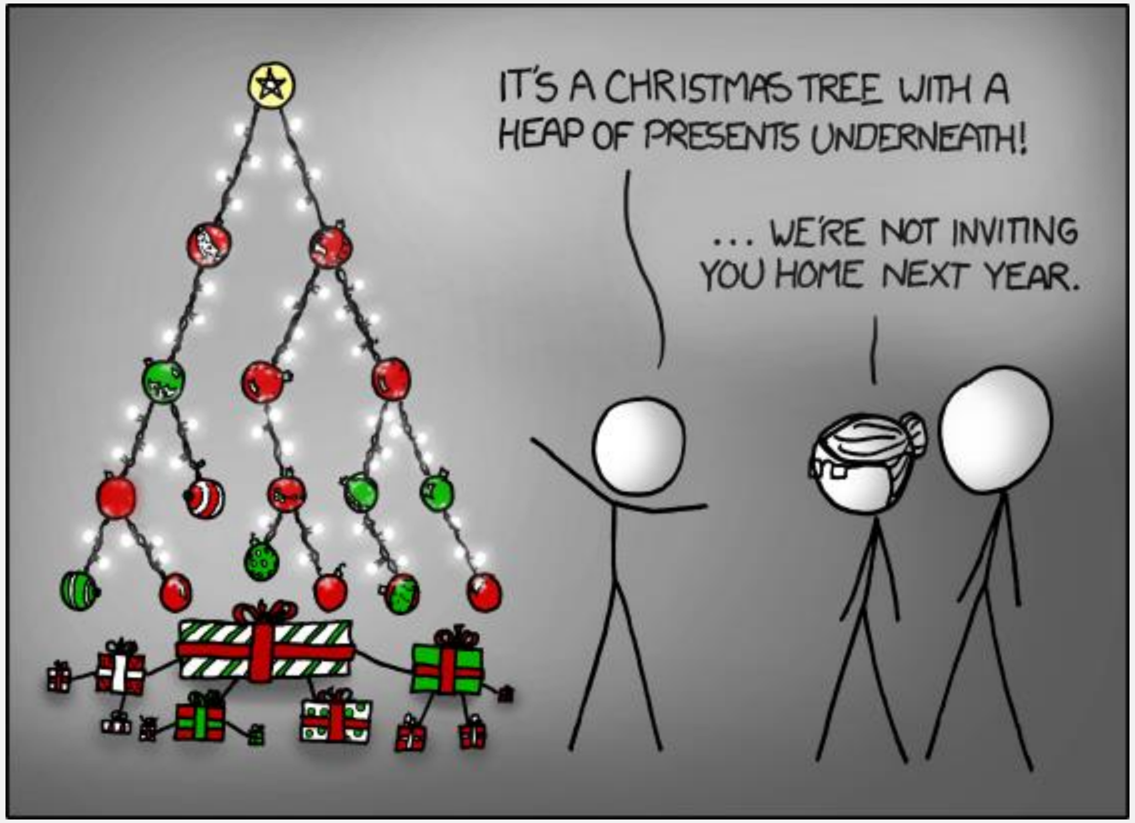
EXACT INSTRUCTIONS

INSTRUCTIONS

ALGORITHMS GONE WILD

- List of famous software bugs: https://en.wikipedia.org/wiki/List_of_software_bugs

WHAT IS A DATA STRUCTURE?

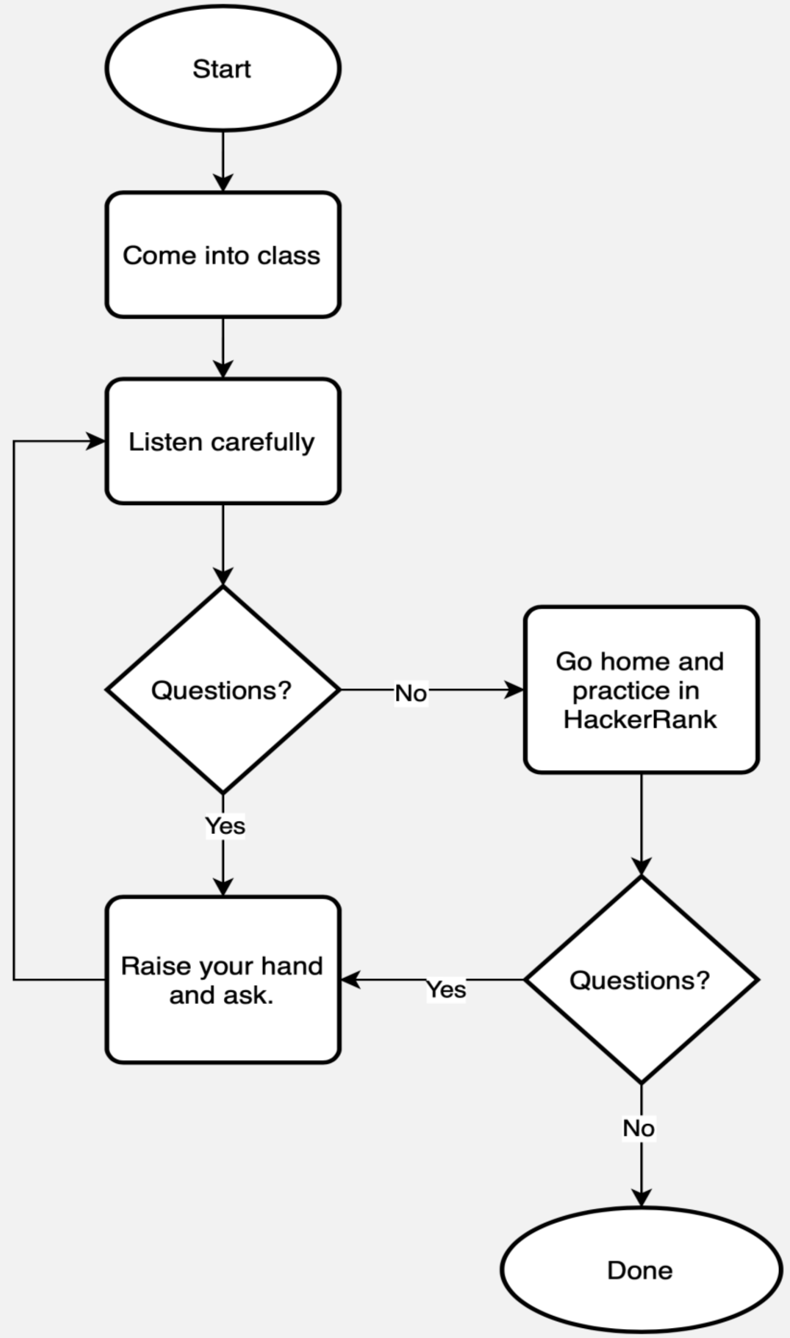

IT'S A CHRISTMAS TREE WITH A HEAP OF PRESENTS UNDERNEATH!
... WE'RE NOT INVITING YOU HOME NEXT YEAR.

A Data Structure is a is a data **organization**, **management**, and **storage forma**t that is usually chosen for efficient access to Data.

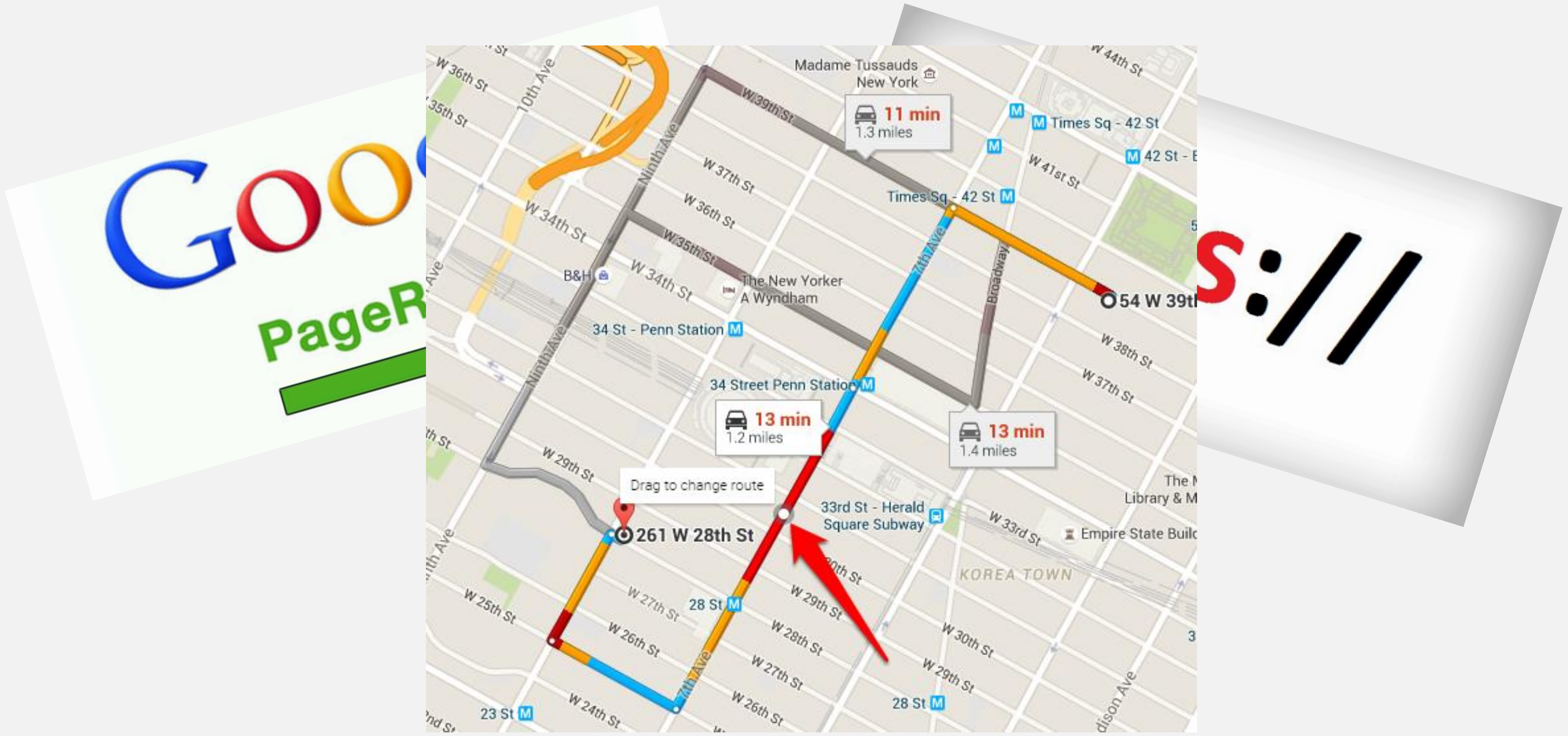
More precisely, a Data Structure is the combination of:

- A Collection of data values
- The Relationships among them
- The Functions or Operations that can be applied to the data values.

OUR FIRST ALGORITHM


Start
Come into class
Listen carefully
Questions?
No
Go home and practice in HackerRank
Yes
Raise your hand and ask.
Yes
Questions?
No
Done

WHAT PROBLEMS DO ALGORITHMS SOLVE?



FACTS ABOUT ALGORITHMS

- They should be *correct*… with some exceptions.
  - Correctness means the algorithm halts at some point with the correct output.
  - The Exception: when we accept erroneous outputs at a given rate (probabilistic algorithms)

- They should be *efficient*… with some exceptions.
  - Efficiency: speed and space
  - The Exception: Problems for which there is no efficient solution possible (NP-complete)

CRACKING THE OYSTER

- Question: "How do I sort a file?" [Jon Bentley. *Programming Pearls, 2nd edition*]
- Precise Problem definition:
  Input: A file containing at most **n** positive integers, each less than n, where **n = 10^7**, and no duplicates.
  Output: A sorted list in increasing order of the input integers.
  Constraints: At most (roughly) a Megabyte of storage is available in main memory.

OUR SECOND ALGORITHM

- Write a python function that receives a list of numbers as input, and returns a new list with these numbers sorted.