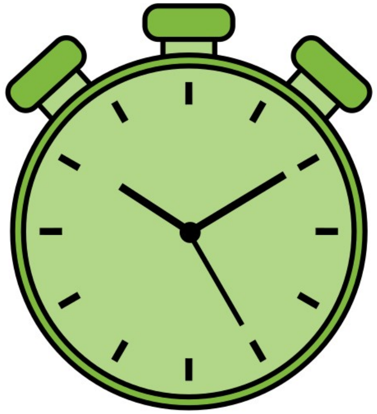
Advent of Code

Dr. Kristian Rother



Approach #1: Contest

5:50	wake up
5:55	coffee
5:56	copy skeleton code from your template
5:59	hit refresh button in browser, repeat
6:00	**GO!**
+X	**Yay!** (X = 10 minutes to 4 days)

Approach #1: Contest

5:50	wake up
5:55	coffee
5:56	copy skeleton code from your template
5:59	hit refresh button in browser, repeat
6:00	**GO!**
+X	**Yay!** (X = 10 minutes to 4 days)

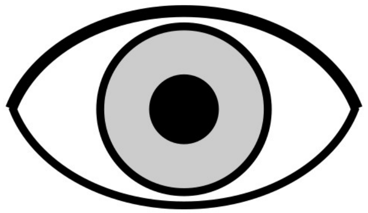
Doing this for 25 days is physically & mentally exhausting!

Approach #2: Sustainable

1. do not rush
2. test your code
3. identify data structures
4. team up

.. make sure you learn something

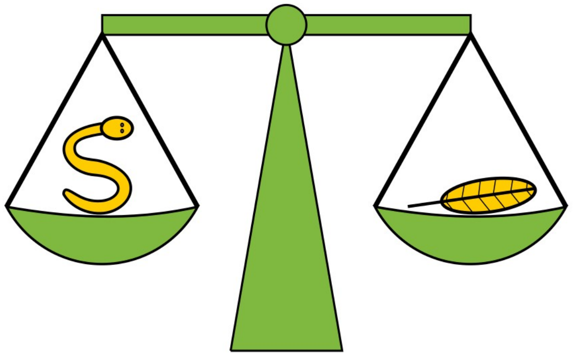
Do not rush



Read the requirements very carefully.

https://adventofcode.com/2021/day/14

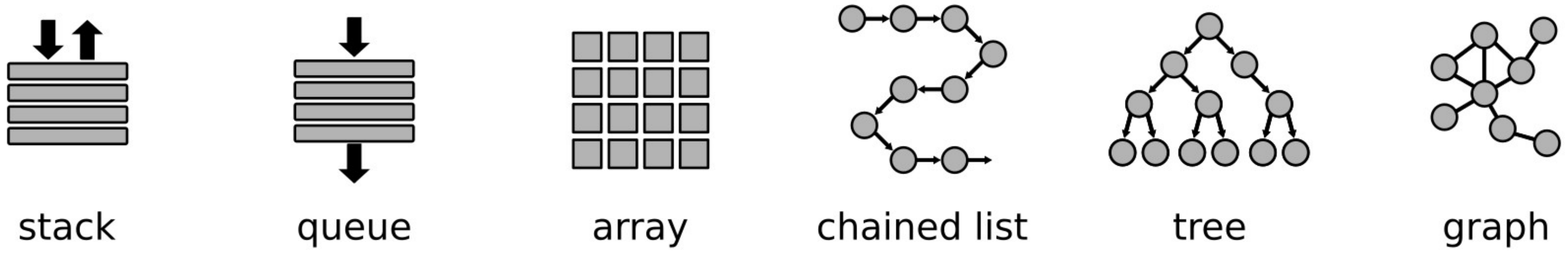
Test your code



- Problem description contains data for Unit Tests
- add more examples with test parametrization
- easy problem: test toplevel function only
- difficult problem: test decompositions, too
- TDD is a great idea

(also see: Uncle Bob, Clean Code Talks, Youtube)

Identify data structures

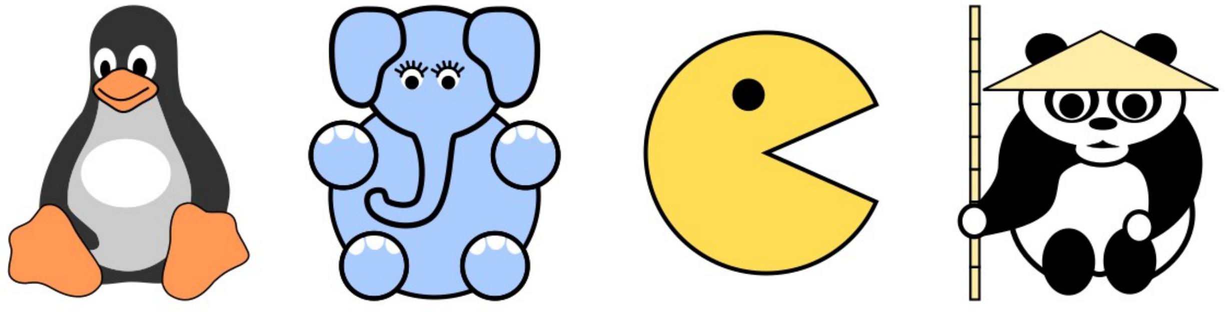


abstraction of data allows you to estimate complexity,

choose an algorithm and, only after that, decide on the implementation.

(also see: Robert Sedgewick, Algorithms, Coursera)

Team Up!



Things that worked:

Pair programming, daily calls, exchange ideas, review code together, having juniors seniors and web devs around

https://github.com/krother/advent_of_code

www.academis.eu