Competitive Programming — Individual Assignment

`http://liacs.leidenuniv.nl/~takesfw/CP`

Spring 2020

This document contains information on the individual assignment of the Competitive Programming course taught at LIACS, the computer science department of Leiden University. Apart from participating in an individual programming contest, the main deliverable is a so-called contest reflection report, in which you are asked to answering various questions. Always give a concise but sufficiently descriptive motivation for your answers. You should do this part of the course individually. Starting from a 1.0, you can obtain 4.5 grade points through the contest itself, and 4.5 grade points through a reflection report, totaling an assignment grade of at most 10.0.

Please hand in your report, typeset using LaTeX, as instructed on the course website. Questions or remarks? Let the course staff know during one of the weekly lectures or lab sessions. If it is urgent, send an e-mail. Good luck!

Individual contest (50%)

Competitive programming in its most practical form involves solving a set of precisely specified problems as quickly and efficiently as possible. This can be done through a programming contest, in which one is asked to solve a number of problems of varying difficulty. Such contests are typically done on-site, with a fixed number of hours of time per team of participants, without access to the internet or external sources other than programming language manuals and a perhaps short team manual.

For this course we will start with a so-called "soft" contest in which you are individually granted a limited number of days to solve a given set of problems. Details on this contest will be announced via the course website. While it is difficult to check the extent to which you use external sources, you are encouraged not to do so unless absolutely not possible otherwise. If you do, explicitly indicate to what extent your solutions were dependent on external sources.

You obtain 0.9 grade points for each solved problem, and can obtain at most 4.5 grade points through solved problems. However, more than five problems will be presented to you in total as part of this individual contest.

Contest reflection report (50%)

In the contest reflection report you are expected provide sufficient detail in order for the course staff to understand your way of working, the extent to which you developed your knowledge and skills and understand your own limitations. You should take a critical look at your own output and way of working, regardless of how you did during the contest itself. Distinguish between contest-related and problem-related aspects, not necessarily in that order.

Contest-related questions

Please answer following questions (worth 2 grade points in total) in your report:

1. How did you prepare for this contest in the past weeks?
2. What was your overall approach to tackling the problems in this contest? For example, how did you decide which problems to solve in which order? How did you process the set of problems? For example, after sorting the problems, did you work “linearly”, or did you return to problems that were difficult at first? Did this work well, or would you do it differently next time?
3. What was your expectation before starting the contest in terms of how it would go? How did it end up going? Why?
4. What knowledge of algorithms and data structures did you miss while working on the contest? How will you be able to obtain this knowledge for future contests?
5. What programming skills did you lack during this contest? How can you train these skills for future contests?

Problem-related questions

For each problem, answer the following questions (worth 2.5 grade points in total):

- What type of problem is this? How did you recognize this?
- If you solved the problem, please answer:
  - What kind of algorithm did you use?
  - What kind of data structures were required?
  - What is the space complexity of your approach?
  - What is the time complexity of your approach?
  - How much time did you spend thinking about this problem, and how much time on coding and debugging?
  - Describe the edge cases of this problem, if any.
  - To what extent did you use external sources to obtain your solution?
- If you did not solve the problem, please answer:
  - Are you able to solve a part of the problem? If so, what were you able to solve, and what were you not able to solve?
  - If you look at the parts that you could not solve, what knowledge or skills would you need to solve these?

End of document. Hand-in details at `http://liacs.leidenuniv.nl/~takesfw/CP`.