Competitive Programming — Course Project (Presentation and Report)

This document contains information on the individual course presentation and report of the Competitive Programming course taught at LIACS, the computer science department of Leiden University. Each student writes two of these reports.

Goal

One way of looking at the course presentation and report, is to see it as a tiny project. The goal of the project is then for an individual student to understand a particular topic related to competitive programming, and to convey this information to the other students such that they also learn about this topic. To this end, each student will give a presentation and write a short report about the topic. The list of topics to choose from can be found at the course website, and includes relevant specialistic a) data structures, b) problem types and c) algorithms for certain problem types.

Each topic should be studied in the context of competitive programming. At least a reference to a book section or chapter is given with each topic. This often includes links to various problems from programming contests related to the topic. Students are strongly encouraged to look up more information on the topic they are assigned. Additional information may be found through literature on Google Scholar, the internet, or even YouTube. Remember that the goal is to master the topic yourself to the extent that you are comfortable explaining it to other students; only reading papers, only showing code or only producing formal definitions may not be the way to go. You should explain the topic in both your presentation and report in such a way that your peer with a computer science background is able to understand it and potentially learn from it, regardless of how experienced that person is at competitive programming.

Topic aspects

Possible aspects to discuss in your presentation and report include (not necessarily in this order):

- *Context.* What is the context of your topic? What other more well-known data structures, approaches and/or more generic problem types is it related to?

- *Relevance.* What is the main distinguishing aspect or feature of your topic? What makes it unique and relevant for competitive programming?

- *Approach.* How does it work? What is the main problem that is being solved? How does this differ from related approaches? Use formal notation where needed and to the extent that it fits your topic.

- *Theoretical aspects.* Discuss at least time and space complexity.

- *Examples.* Give at least one specific example of how your topic finds its way into competitive programming. It is strongly recommended to consider having a running example in your work.

- *Take-home messages.* What are the "do's" and "don'ts" with respect to this topic? When can it be used in a competitive programming context? How does one "recognize" whether it is applicable to a certain problem?

Depending on the type of topic you are studying, one or a few of the above aspects may not apply. Remember that the goal is not to be exhaustive in explaining all the above, but to educate your peers on your topic.

Presentation

Each student gives a presentation of around 25 minutes, followed by at most 10 minutes of discussion. The goal of the presentation is to teach the audience about your topic. The discussion should be used to constructively talk with fellow students about detailed aspects of the topic. Remember that if you also have code examples, 20 slides is already a lot. From your presentation, and perhaps after some practice, fellow students should learn how to apply your topic in their future competitive programming endeavors.

Report

The report on your topic, for which there is a strict deadline near the end of the course, should be written in LaTeX, using 4 to 6 concisely written pages. Use a default style:

```
\documentclass{article}
\usepackage[a4paper, total={14cm, 23cm}]{geometry}
```

The same aspects as in the presentation should be covered, but now in a written format, which may at some points require a slightly different form than a presentation.

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