Datastructures and Algorithms Lab

Saska Dönges

2.9.2020

Course assistants

- Course assistants
  - Saska Dönges – **saskeli** – first-name.last-name@helsinki.fi
  - Hannu Kärnä – first_name.last_name@helsinki.fi
- Contact via e-mail or the course telegram channel
- Telegram channel https://t.me/tkttiralabra/ (will contain discourse in Finnish)
- Course page https://tiralabra.github.io/2020_p1/en

What?

- On the course you will implement some program that utilizes "difficult" algorithm(s)/data structure(s) including all of the required data structure and algorithms

Prerequisite knowledge

- Prerequisite knowledge:
  - **Data Structures and Algorithms**
    - Mandatory
  - **Programming techique** (only in Finnish)
    - Very beneficial but the course materials cover the required knowledge in testing and project sturctures
- Ask if:
  - You have done Data Structure and Algorithms but do not have the credit (missing the exam for example)
  - You are unsure about the prerequisite knowledge requirements

Course content

- The programming language to be used has to be accepted by a course assistant. At least Java will be accepted
  - Ask a course assistant if you prefer some other languages
  - Note that testing and test coverage reporting is required no matter the language
- **Git** version control and **GitHub** is used
- The lab consists of **individual work**
- Generally the product will be some sort of **running program**
  - Not a library or a bunch of code that can not be executed
- Typically progams need to have some sort of **user interface**

Course content

- Project examples:
  - Comparison of path finding algorithms
  - Data compression algorithms
  - Cryptography
  - Game solvers (minesweepers, nonograms, solitaire...)
    - There is a minesweeper template project on git!
  - AI (Chess, Go...)
    - There is a chess template project on git!
- **Try to chose a topic you are personally interested in!**

Coding style

- The code written for this course should be of high quality and easy to read. You should use some kind of style checkking (e.g. Java Checkstyle)
- The project name should be indicative of the contents. Course assitants will not appreciate it if all the project names are along the lines of “Lab-2020”
- Project structure along the lines of the programming techniques course
  - I.e. not all of the project code should be in the same file/folder
- Good coding conventions like DRY ja Single responsibility should be applied

Conduct of the course

- **Deadlines** according to the course material
  - Each deadline gives 0-3 points based on the deadline requirements
    - The first week deadline only gives up to 1 point
  - Submissions are done by *pushing* the project state to GitHub
    - No submissions by e-mail
  - A large part of the points – and the grade – are based on deadlines
  - After each deadline, a course assistant will give some sort of feedback – more thorough feedback is available through paja, e-mail or Telegram
  - Extra time for deadlines may be available with **good reasons** if asked for **in advance**.

Conduct of the course

- Weekly workshopw (paja) is available on mondays and tuesdays on zoom according to the timetable
  - Mondays 16-18 and tuesdays 14-16, there will be help avaialable in English
  - Personal guidance can be provided on campus or Zoom if requested
  - Not mandatory
  - **Best way to get help from a course assistant**
  - Telegram is not an official source of information but can be useful
  - Other TAs in paja may also be able to help even if the lab is not explicitly on the agenda at the time

Conduct of the course

- A **code review** is done in conjunciton with Deadlines 4 and 5
  - Every student will get another student project to review
  - Students write and receive feedback on each others' projects
  - The intention is to get familiar with reading code written by others
  - Maximum points for each review is **2**

Conduct of the course

- At the end of the course there will be a mandatory **demo session**
  - Each student presents their project for 3 to 5 minutes
  - Every student is present for the entire session
  - The project does not have to be completely done at the demo session
- There is **no course exam**

Motivation

- This can be one of the most fun courses during Bachelors' studies - you can implement whatever you want!
- If you get stuck, ask a course assitant for help
  - I'm here for you!
- Normally there are no real penalties for dropping courses – labs are an exception to this
  - It may be harder to enroll to the course after dropping the course

Ad break

- During fall of 2019 a group of students created 2 new project templates for the lab
  - Chess and
  - Minesweeper
- If either of these subjects are of interest, you may want to check them out. Links can be found on the course page

Thank you!

- **Welcome to the course!**
- Most information about the course can be found at: https://tiralabra.github.io/2020_p1/en
  - It's a good idea to read through the entire site!
- I will stay for a while to answer any questions you may have!