Data Structures Lab Course (start of summer 2023)

Weekly report 1

Learnings and tasks for week 1

This week I have completed below tasks:

- went through the course materials in Finnish.
- programming language Python selected for the topic. The project was created via Visual Studio.
- created a GitHub public repository for the project and then registered with Labtool. Enabled adding issues to the repository.
- pushed local files or modifications to GitHub repository via Git Bash.
- learned 4 new algorithms: JPS, IDA*, fringe search, A*. IDA* was selected as the main algorithm.
- completed the Specification Document, choosing the topic "implement a solution to find the shortest distance path between two street addresses".
- completed the weekly report 1 for week 1.

In the beginning of the first week, I had some pressure to choose from topic options. From the topic idea list, a lot options seems to be good choice, like the topic of graphs and pathfinding (verkot ja polunetsintä), the topic of games (pelit), or the topic of data compression (tiedon tiivistys). Graphs and pathfinding seems can be implemented in Python language, but the challenge seems to be producing a map/labyrinth visualization. Therefore, I learned something new by searching from internet and learned what ASCII graphics and visualization are. (http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html)

The topic selected is to implement a solution to find the shortest distance path between two street addresses.

What would be required for this program's structure? A basic User Interface would be needed, which can take input (start point's address and destination address) into program. The core is to identify efficient data structures and algorithmic techniques to find shortest route between two street addresses. Dijkstra's algorithm discussed in the Data Structures and Algorithms course is a good option. It means it needs to transfer the road network into a weighted graph, and go through a massive amount of nodes and weighted edges before being able to identify the shorted path. Therefore, I need to evaluate whether Dijkstra's or other algorithm is a good option when considering time complexity and space complexity requirements.

After that, applying the selected algorithm to find the shortest distance is possible. Once the shorted path is found, need to calculate the total distance between the starting address and destination address.

What was unclear or caused difficulties?

Problem 1:

How I can choose or know which would be an appropriate shortest path algorithm for this project?

Problem 2:

I plan to obtain the road network from the online map service Open Street Map
https://www.openstreetmap.org, but I don't know yet how to convert the map into a graph, which I need to clarify in next week. It's also critical to consider how I can visualize the routes (or can present the questions in this way: how to implement visualization of the path by covering nodes and jump points at early stage of development), and how I can convert street address into a data format which supports the algorithms' operation and supports calculating the distance. I am aware of that Weighted Graph data structure and Shortest Path algorithms learned from the Data Structures and Algorithms course would be helpful for this project, but I am not sure what other data structures and algorithmic techniques will be needed to solve this project's problem.

Problem 3:

How I can check the correctness of algorithms? How I can thoroughly test the solution, by using different start and end addresses.

Working hours

Week 1		
Date	**Hour**	**Content**
15.5.2023	2	read through course materials
16.5.2023	4	not know which topic to choose, study new algorithms: JPS, IDA*, fringe search & A-star
19.5.2023	4,5	consider the content for specification, what required for implementation, achieved basic design.
20.5.2023	4	Documentation (weekly report, specification)
21.5.2023	6	IDA*, specification
Total	20,5	