ASSIGNMENT 2 REPORT

LAMIJA IMAMOVIĆ

Approaching this assignment, I didn't expect it to be as tricky as it turned out. Due to time constraints, some parts are incomplete, especially the constraints task, which didn't quite work out. The testing part is done, but there's uncertainty if it aligns perfectly with the assignment requirements. Despite this, I tackled everything to the best of my understanding and ability. In the midst of challenges, understanding and applying Dijkstra's algorithm was tough. But, the other parts of the assignment, like creating classes such as Graph, Node, ShortestPathSolver, and Constraints, went relatively smoothly. The Graph class is crucial for representing a graph's structure. It holds nodes, edges, and methods to manipulate the graph. Represented as an adjacency list, it helps with tasks like identifying neighbors and adding edges efficiently. The Node class contains vital info about a node, like short and full names, neighbors, and methods to work with this info. Nodes are like the corners of the graph and are essential for the Graph class. ShortestPathSolver manages finding the shortest paths in a graph, considering constraints. It uses Dijkstra's algorithm with tweaks and has methods to print shortest paths, apply constraints, and find the shortest path between specific nodes. Constraints introduce the idea of limitations on edges. It includes a starting node, ending node, constraint type, and probability. The probability decides if an edge should be turned off based on a random value. Unfortunately, I couldn't get this part right. Places class analyses places provided in txt file and makes it easier for usage in other classes. The tests act like experiments for the code, checking if paths are right, seeing how constraints affect results, and looking at special cases like loops and negative weights. In looking back, despite the challenges and acknowledging the incomplete parts, this assignment gave me useful insights into graph theory, problem-solving with algorithms, and the importance of thorough testing for strong code performance. I would definitely change my approach to task and do it earlier, since I have gotten myself in situation to do final checks on New Years Eve. I know that I will not get all the points, but at least I have done it myself and now I can hope for 6/7 points :)). To be honest, I have no idea what more to write, so I will finish it here. Paths are right, seeing how constraints affect results, and looking at special cases like loops and negative weights. In looking back, despite the challenges and acknowledging the incomplete parts, this assignment gave me useful insights into graph theory, problem-solving with algorithms, and the importance of thorough testing for strong code performance. I would definitely change my approach to task and do it earlier since I have gotten myself in situation to do final checks on New Years Eve. I know that I will not get all the points, but at least I have done it myself and now I can hope for 6/7 points :)). To be honest, I have no idea what more to write, so I will finish it here.