Assignment Review

The proposed assignment tackles the perennial issue of urban transportation in Sarajevo by introducing a unique solution involving quantum mechanics-inspired glass capsules. This review reflects on the implementation process, challenges faced, and the effectiveness of the designed program for finding optimal paths between various city points. The tasks were interconnected which made coding easier, since every additional code was just an “upgrade” on the already existing code.

The first task was to create a weighted graph, which I decided to do using a map in Graph class which contained nodes from the Node class. This way I found to be the most logical implementation and I did not experience any massive issues while writing the code. Apart from that, I believe that it makes my code look clean and readable. Inside the Graph class, I have ensured that in the creation of the edge, i.e. node, the constraint that might exist between the two nodes processed at the time is accounted for and the time/distance is correctly calculated with the appropriate delay. That is exactly what the fourth task described to be done.

The second task was to create a class for solving the shortest path algorithm between any two nodes in the graph, with proper exception handling. I decided to implement Dijkstra algorithm and that is something that took the most time to do. Luckily the problems were not major and easy to solve, with the help of the Internet and YouTube. Some of the exceptions that I needed to handle were situations where there is no path between the two nodes, situations where there are negative distances and situations with non-existent nodes. All these situations are also covered in tests, and I have also tested the algorithm for self-loops and whether all nodes in the graph are connected amongst themselves, which third task noted to be done in the main program as well.

To conclude, I would say that I am satisfied with the overall implementation and solution of the given assignment. It had an interesting storyline to it and made the “dull” task of Dijkstra implementation on the graph fun. As far as the future improvements are concerned, I personally do not know what exactly would be significantly better to do, but there is always a better way to do something, so I am sure that something could be done here as well.