Burak Bozdağ
150170110

BLG336E – Analysis of Algorithms II Homework 2 Report

Problem Description

Joseph and Lucy are best friends and they have arranged a travel together two months ago and reserved their flying tickets. However unfortunately, in these two months, they have fought over a small issue and they are not in good terms, so do not want to face each other in any case. Even so, they did not cancel the travel since everything was planned before. They decided to do check-in separately and stay at different hotels during the travel. Also, when going around the city, they do not want to run against each other.

- The city map is represented as a **directed** and **weighted** graph where nodes are places and edges are roads. Edge weights are the lengths of roads.
- Two of the nodes are Joseph's hotel (JH) and Lucy's hotel (LH). The other two of the nodes are the destination for Joseph (JD) and destination for Lucy (LD).
- If there is a path from node A to B with length of n, then it takes n minutes to get to B from A.
- Joseph and Lucy leave their hotels at the same time and they must not be in the same node at the same time.
- They spend 30 minutes at the destination before returning their hotels.

I found the shortest paths from hotels to destinations and destinations to the hotels that are not intersecting considering the instant locations of Joseph and Lucy.

PART 1 – Constructing the Graph

I used the adjacency matrix representation for the graph construction. The txt file for graph input can be given as a command line argument in this form: *a.out test1.txt*.

PART 2 – Implementing the algorithm

I implemented a shortest path algorithm (Dijkstra) using C++ language. I found 4 shortest paths for each input as:

- **JH to JD** and **JD to JH**,
- **LH to LD** and **LD to LH**.

Steps of the algorithm is like this:

- Find the shortest paths from hotels to destinations.
- Check if there are any case that Joseph and Lucy are at the same node considering the durations.
- If there is an intersection, find an alternative path which is valid and shortest. You should try to change either Joseph's or Lucy's path and choose the better one. If there are no alternative paths for both of them, then print "No solution!" on the screen and terminate the program.
- Then, find the shortest paths from destinations to hotels in the same manner. Note that they have waited at the destination for 30 minutes.

Sample running of the program: