Rüzgar Ayan 21801984

2020 FALL CS473 Programming Assignment 2 Report

Approach for part (a)

Part (a) is simply an instance of single-source shortest-path problem. As told in the assignment, I have implemented and used Dijkstra algorithm with a min-priority queue. I used my min-heap implementation that supports buildHeap operation so that the Dijkstra will run even faster. There is not much to talk about the solution since I just implemented the Dijkstra as known by everyone. However, it is worth to note that I used the compact form explained in the lectures to store the graph data structure. And this operation of parsing the ".mtx" file into a graph required me to implement a Linked List data structure and this was actually as hard as solving the actual given problem.

Changes for part (b)

Instead of minimizing the **sum** of the weights as in a path in part (a), we are maximizing the **product** of the weights in a path for this part. This firstly requires the usage of a max-priority queue instead of a min-priority queue. Also, we must change our relaxation conditions according to this problem so that we are trying to achieve a higher probability path. Other than changing the underlying data structure and a few lines in the relaxation, part (b) is really similar to part (a).

Correctness of part (c)

What we do in part (c) is actually converting the problem (a) to problem (b) by using the properties of logarithm function. When adding the logarithms of two weights in part (a)'s algorithm, we can actually multiply the original weights because of the property $\log(a) + \log(b) = \log(ab)$. Since we also have a minus (-) sign in front of the logarithm when converting, this allows us to use again a min-priority queue instead of max-priority queue. Because finding the maximum of a set of values is equivalent to finding the minimum of the negatives of those values. This comparison with the logarithms of two number actually works because of the fact that the logarithm function is monotone.

In the end, we convert the logarithm weights back to the real weights by exponentiating 2 and this is obviously correct as logarithm and exponentation are inverses of each other.