Report for Long Project 3

Shortest Path Algorithms

Implementation of Data Structures and Algorithms

By G16

Team Members:

Vasudev Karthik Ravindran

Ian Laurain

# Introduction:

The aim of this project is to implement the various shortest path algorithm and understand the working aspects of the same. This document gives a detailed description of modules used in the project.

# Methodology:

In level 1, the shortest path algorithms namely BFS, Dijkstra's algorithm, DAG shortest paths, and Bellman-Ford algorithm are implemented as BFS.java, Dijkstra.java, DAGShortestPaths.java and BellmanFord.java respectively. Driver Program for level 1 is LP3DriverLevelOne.java file where if all edges are of same weight, then BFS algorithm is performed; if graph is a DAG, DAGShortestPaths algorithm is performed; if all edges have positive weights, Dijkstra’s algorithm is performed; else Bellman Ford Algorithm is performed. If Bellman ford returns null (in case of negative cycle), then statement “Dijkstra's algorithm, DAG shortest paths, and Bellman-Ford algorithm” is displayed else shortest path details are displayed as per the specification.

In level 2, the number of shortest paths in a graph from a source vertex is implemented using the following approach:

1. Graph vertices are relaxed using the level 1 shortest path implementation.
2. Then we create a subgraph of G by adding all the edges of G which satisfy the following condition:

For each edge (u, v) of G, if v.distance = u.distance + e.weight, then add edge to subgraph.

1. Apply the BFS on the subgraph, to update the individual path counts of each vertex.
2. Get the sum of all individual path counts of vertices.
3. If the graph has negative or zero cycles, edges of the cycle are found by storing edges of cycle as parent in Bellman-Ford algorithm; for zero cycles detection, the class HasDirectedCycle.java is used.
4. Then the output is printed according to the given specification.

# Test Results:

All the sample test cases provided as part of this project have run successfully and have been verified to give same output as expected.

# Conclusion:

Thus the shortest path algorithms are implemented and verified successfully. We can also understand about the different conditions which are suited for different projects.