

Graph Algorithms Worksheet

Dijkstra's Algorithm

Goal: Find the shortest path from some start vertex to all other vertices in the graph.

Runtime: $O((|V| + |E|) \log(|V|))$

Pseudocode

- **Initializing data structures**
 1. Initialize the following structures:
 - Fringe
 - DistanceMap
 - PredecessorMap
 2. Add the start vertex to the fringe and the DistanceMap with distance 0
 3. For all other vertices, add them to the fringe and the DistanceMap with distance infinity
- **While-Loop (processing the shortest paths)**
 1. Pop off a vertex v from the fringe
 2. Loop over each neighbor n of v :
 - (a) Let `newDistance` = `DistanceMap[v]` + `edge(v, n)`
 - (b) If `newDistance` < `DistanceMap[n]`, then
 - (1) Update the priority value of n to be `newDistance`
 - (2) Update the predecessor map so that the parent of n is v

Lab Specific Notes

- Remember that the `Vertex` in our lab is actually just an `Integer`
- The Java `Priority Queue` does not have a `changePriority` method. So, you cannot actually put all vertices into the fringe in the initialization step, then update their value as needed. Instead, consider either
 - (1) adding/removing a vertex from the PQ every time you "update", or
 - (2) adding the vertex to the PQ as usual but keeping a `visited` set.
- Your `shortestPath` method should be able to return the optimal path from the start vertex to the goal vertex. Thus, you should be able to reconstruct that path using the predecessor map.

Practice Problems

Consider a weighted, undirected graph. If the question is T/F and the statement is true, provide an explanation. If the statement is false, provide a counterexample.

1. (T/F) If all edge weights are equal and positive, breadth-first search starting from node A will return the shortest path from a node A to a target node B.
2. (T/F) If all edges have distinct weights, the shortest path between any two vertices is unique.
3. (T/F) Adding a constant positive integer k to all edge weights will not affect any shortest path between vertices.
4. Draw a weighted graph (directed, or undirected) where Dijkstra's would incorrectly give the shortest paths from some vertex.

Answers available at <http://datastructur.es/sp17/materials/discussion/discussion12ep.pdf>