Assignment 22

Alex Clemmer
Student number: u0458675

1

Really simple. You reverse G and then check that for any node $v \in G$, `prev[u] = v`. In other words, `prev[]` gives us the links to the last nodes, and by reversing G, we can check if these edges are valid. Since reversing G is linear, and since this is linear, net time should also be linear.

2

Since we've checked the edges in the previous step, we know that all nodes in `prev[]` are also $\in G$. Since they encode the shortest path, we know that each node has at most one parent. Thus, in order to find out whether the nodes are V, all we have to do is count the number of nodes in each. If they are the same, we have a spanning tree.

3

The very first thing we need to do is find the distance of at every node. Since our spanning tree guarantees that we visit every node, and since no node has more than one parent (and therefore, that every node has exactly one path leading to it), and since this tree is rooted, conceptually what we would hope to do is to start at s and simply add the weights up for each node we encounter after that.

In practice, things are slightly more complicated. We start by reversing `prev[]` as described in previous assignments, which is a linear operation. Then we add the weights up in the tree. This is also linear.

Now the interesting part: since we have a distance for every node in the graph, every edge in G that is not in our spanning tree goes from a node that we have *already given a weight to*, to another node, which *also must have a weight given by us.* So the impact is, if the total weight of a given missing edge plus the distance of the node it's coming form is less than the distance of the node it is going to, the graph breaks.

This leads us to an incredibly simple conclusion: just check every edge in G. If the edge is missing, find the weight of the edge w + the distance at the node we're traveling from. If it is greater than the total distance of the node it's traveling to, then our spanning tree is not the shortest distances tree. If we've exhausted all edges and found no contradicting evidence, then the spanning tree is the shortest distances tree.