

## Yingzhu (Jacqueline) Zhang

1. 4.2.9 **Solution:**

```
public boolean is_topological
{
    for (i=0; i< number of vertices in the permutation of A DAG; i++;)
    {
        mark  $vertex_i$ ;
        for (n= 0, n< outdegree of the current vertice; i++)
            if the outdegree vertex is marked
                return false
    }
    return true;
}
```

2. 4.2.11 **Solution:**

Consider a collection of digraphs of one center vertex with directed edges to and from it and every other outer vertex, starting with a graph of two vertices. This collection is exponential because as you add additional outer vertices, the number of cycles will increase by  $2v$ , where  $v$  is the number of additional vertices.

\* See attachment for demonstrative diagrams.

3. 4.2.27 **Solution:**

The BFS does not necessarily produce a topological order. We explain this by examples. Consider the following cases:

Case 1: The source points towards multiple vertices, among which at least one points to the other. This causes an issue, for all vertices are listed as the same distance from the source, and the fact that one is pointed to by another is not taken into account.

Case 2: There are multiple sources that would cause multiple numbering schemes. In this instance, two vertices may be marked as the same distance from their respective sources even though one points to the other as well.

\* See attachment for demonstrative diagrams.

4. 4.3.4 **Solution:**

*Counter – example:*

Consider the demonstrative diagram, which is a edge-weighted graph. Some edge weights contained are different, but not all edge weights are distinct. Using both Prim's algorithm and Kruskal's algorithm, it can be seen that there

is only one MST, A-B-C-D, despite the fact that the edge weights are not all unique.

\* See attachment for demonstrative diagrams.

5. 4.3.8 **Solution:**

*Proof* by contradiction:

Consider that you have a MST of a graph containing the edge of maximum weight of a cycle  $E$  within the graph.

Now, it is a property of trees that, if you remove an edge from a tree, the graph is no longer connected.

Also, a tree contain no cycles.

It follows that if you remove  $e$  from the MST, there is some other edge,  $e'$ , within the cycle that is not currently in the MST and will reconnect the MST. By definition of  $e$  being the edge of maximum weight in the cycle, the weight of  $e$  will be less than the weight of  $e$ .

Thus, the edge of maximum weight in a cycle does not belong in the MST of the graph containing the cycle.

$\therefore$  Given any cycle in an edge-weighted graph, the edge of maximum weight in the cycle does not belong to the MST of the graph.

6. Find a minimum weight spanning tree for the graph below using both Prim's and Kruskals algorithms. Is the spanning tree unique? Why or why not?

**Solution:**

\* See attachment.