Assignment 1: Minimum Spanning Trees

Due: 2nd week (submission through moodle before sessional class)

Minimum Spanning Tree (MST): Given a weighted undirected graph, a minimum spanning tree is tree of minimum weight that connects all the vertices.

Second Best Minimum Spanning Tree (SBMST): Second-best minimum spanning tree is a spanning tree of minimum weight that has at least one edge that is not in the (best) minimum spanning tree.

Input: First line and second line of input file will contain the number of vertices, *n* and number of edges, *m* respectively followed by *m* lines each containing source, destination and weight of an edge.

For example:

```
6
9
1 2 4
1 3 1
1 4 3
2 3 4
2 4 4
3 4 2
3 6 4
4 6 6
5 6 2
```

Section B1
Implement Prim's algorithm for finding minimum spanning tree of a graph. Start from vertex 1 and whenever there is a choice of vertices, pick the vertex with smallest number. Use the graph data structures and heap you implemented during CSE204. Your algorithm should run in time O(*E*lg*V*).

Output: Weight of MST as well as the edges in the MST. Sample output for the input above is

```
1 3
3 4
1 2
3 6
5 6
13
```

Section A1
a) Implement Prim's algorithm for finding minimum spanning tree of a graph. Start from vertex 1 and whenever there is a choice of vertices, pick the vertex with smallest number. Use the graph data structures and heap you implemented during CSE204. Your algorithm should run in time O(*E*lg*V*).

b) Implement an algorithm to find a second best minimum spanning tree of the given graph.

Output: Weight of MST and SBMST as well as the edges in MST and SBMST. Sample output for the input above is

1 3
3 4
1 2
3 6
5 6
13

1 3
3 4
2 4
3 6
5 6
13

Section B2

a) Implement Prim's algorithm for finding minimum spanning tree of a graph. Start from vertex 1 and whenever there is a choice of vertices, pick the vertex with smallest number. Use the graph data structures and heap you implemented during CSE204. Your algorithm should run in time O(Elg$V$).

b) Implement an algorithm to find a second best minimum spanning tree of the given graph.

Output: Weight of MST and SBMST as well as the edges in MST and SBMST. Sample output for the input above is

1 3
3 4
1 2
3 6
5 6
13

1 3
3 4
2 4
3 6
5 6
13

Section A2

a) Implement Kruskal's algorithm for finding minimum spanning tree of a graph. Use the graph data structures you implemented during CSE204. Your algorithm should run in time O(Elg$V$).

b) Implement an algorithm to find a second best minimum spanning tree of the given graph.

Output: Weight of MST and SBMST as well as the edges in MST and SBMST. Sample output for the input above is

1 3
3 4
5 6
1 2
3 6
13

1 3
3 4
5 6
2 4
3 6
13