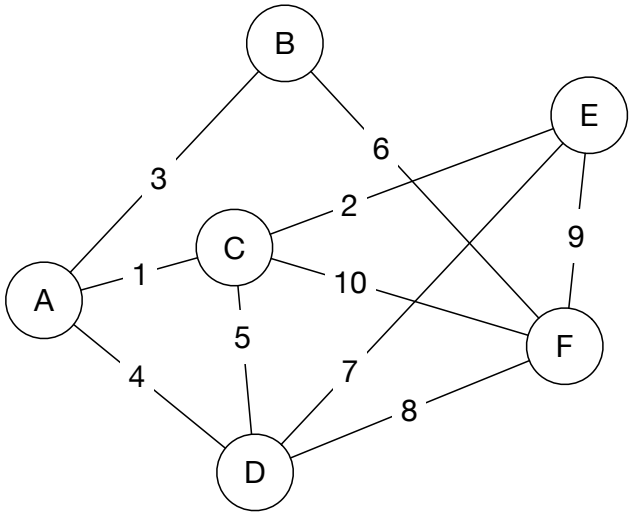
Name & Student Number:

1. Find a Minimum Spanning Tree (MST) in the following graph. (Highlight edges in your MST.)



2. Answer each of the following questions as True or False, and in one line justify your answer.
   (a) A MST contains a cycle.

   (b) If we remove an edge from a MST, the resulting subgraph is still a MST.

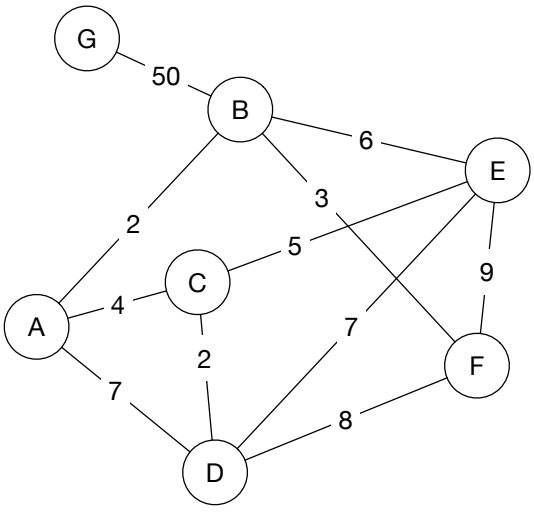
   (c) If we add an edge to a MST, the resulting subgraph is still a MST.

   (d) If there are V vertices in a given graph, a MST of that graph contains $|V|$ edges.

3. Following is the pseudocode for Prim's algorithm to find a MST.

```
1: function PrimMST(Graph G)
2:     initialize distances to ∞
3:     mark source as distance 0
4:     mark all vertices unprocessed
5:     for each edge (source, v) do
6:         v.dist = w(source,v)
7:     end for
8:     while there are unprocessed vertices do
9:         let u be the closest unprocessed vertex
10:        add u.bestEdge to spanning tree
11:        for each edge (u,v) leaving u do
12:            if w(u,v) < v.dist then
13:                v.dist = w(u,v)
14:                v.bestEdge = (u,v)
15:            end if
16:        end for
17:        mark u as processed
18:    end while
19: end function
```

(a) Find a MST in the following graph by running the above Prim's algorithm, starting at vertex A. Fill the table.



Vertex	Distance	Best Edge	Processed
A			
B			
C			
D			
E			
F			
G			

(b) Do a runtime analysis of Prim's MST algorithm. (Annotate the above pseudocode with your runtimes.)