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.

```
function PrimMST(Graph G)
    initialize distances to ∞
    mark source as distance 0
    mark all vertices unprocessed
    for each edge (source, v) do
        v.dist = w(source,v)
    end for
    while there are unprocessed vertices do
        let u be the closest unprocessed vertex
        add u.bestEdge to spanning tree
        for each edge (u,v) leaving u do
            if w(u,v) < v.dist then
                v.dist = w(u,v)
                v.bestEdge = (u,v)
            end if
        end for
        mark u as processed
    end while
end function
```

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

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