

**CSC 323 Algorithm Design and Analysis, Spring 2020**  
**Instructor: Dr. Natarajan Meghanathan**

**Assignment 12: Prim's Minimum Spanning Tree Algorithm**

**Due: April 21st, 2020** (11.59 PM, in Canvas).

In this assignment, you will run the Prim's algorithm to determine minimum spanning trees on a graph assigned to you. You should illustrate the working of the algorithm through a sequence of iterations (as shown in Example Graphs 1 and 2).

**Submission:** You could either draw in hand and scan the solution to a PDF file and submit or draw in Power point and print/save the Power point slides to a PDF file and submit.

**The high-level description of the Prim's algorithm is as follows:**

Step 1: Initialize the tree with a single vertex, chosen arbitrarily from the graph. Initialize the set of *candidate edges* with the edges connecting the vertex in the tree to the vertices that are not yet in the tree.

Step 2: Grow the tree by one edge: Among the edges (referred to as the *candidate edges*) that connect the vertices in the tree to the vertices that are not yet in the tree, find the edge of the minimum weight and transfer it to tree (if two or more candidate edges are of the same minimum weight, break the tie arbitrarily).

Step 3: Update the set of candidate edges as follows:

Include the edges (if any) that are between the recently included vertex in the tree (from Step 2) and to vertices that are not yet in the tree

Exclude the candidate edges (if any) that are between the recently included vertex in the tree (from Step 2) and to vertices that are already in the tree.

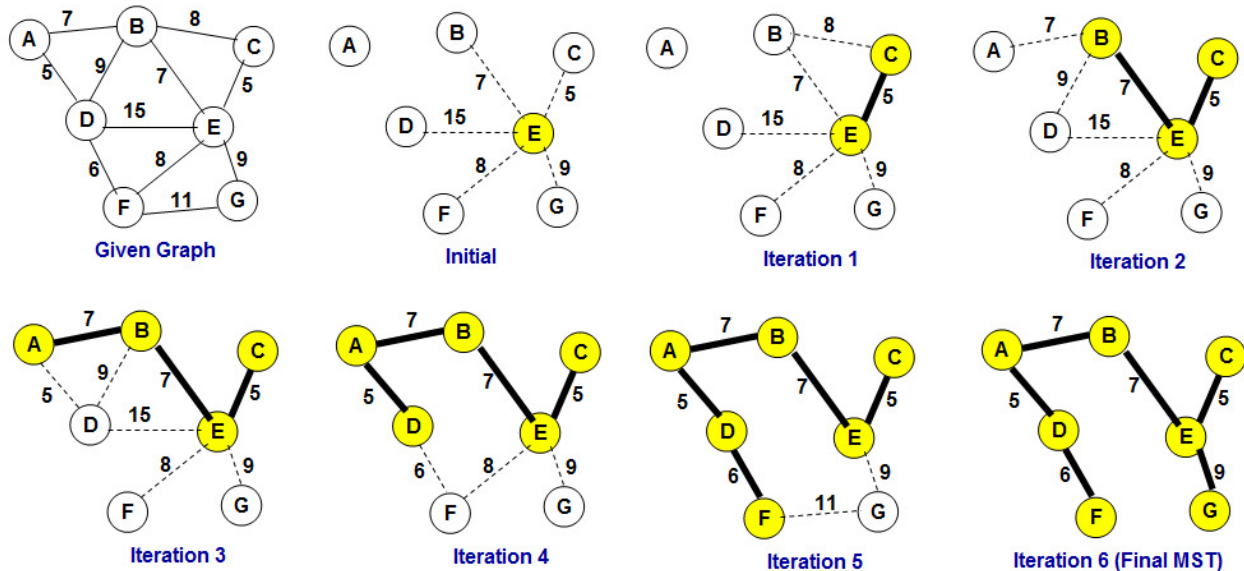
Step 4: Repeat Steps 2 and 3 until all the vertices in the tree.

The working of the above algorithm is illustrated below on two example graphs.

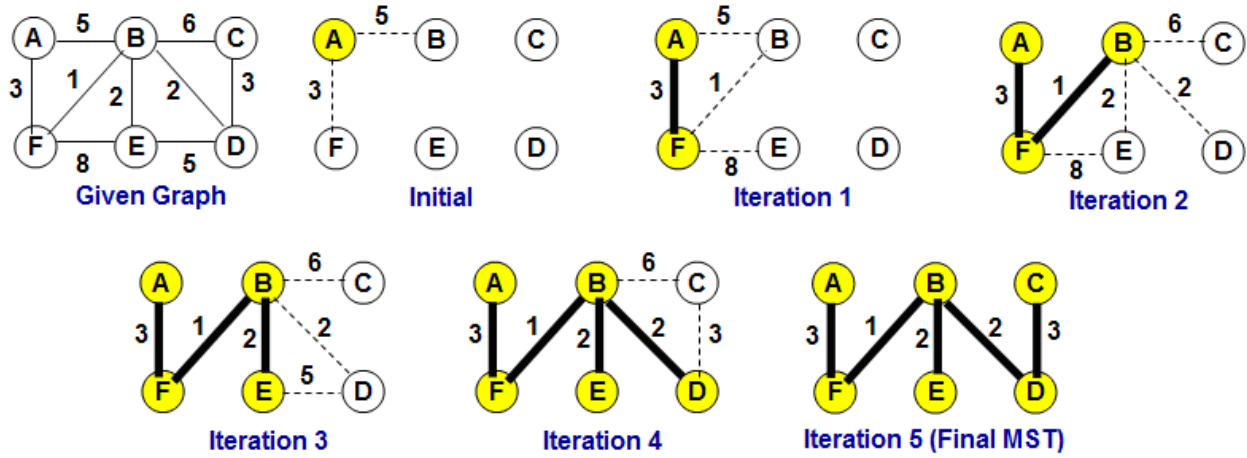
In each iteration, the candidate edges are shown in dotted lines and the tree edges are shown in bold lines. Also, the vertices that are included in the tree are highlighted in yellow and the vertices that are not yet in the tree are not highlighted.

You should show similar work on the graph assigned to you.

**Example Graph 1**

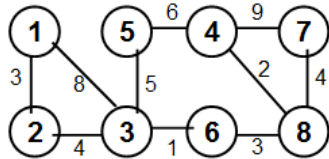


**Example Graph 2**

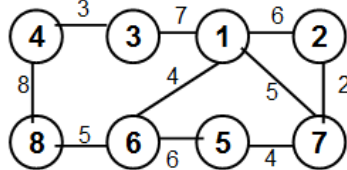


**Graphs Assigned to Students**

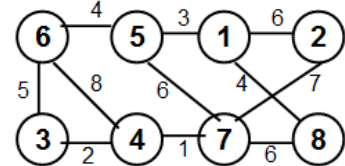
Abate, Biruk



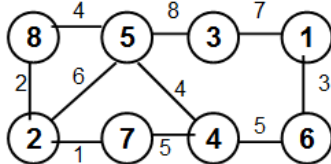
Akintade, Oluwaseun



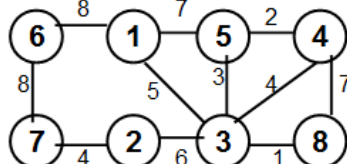
Alharbi, Abdullah



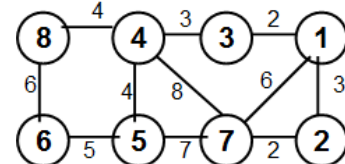
Alharbi, Abdulmajeed



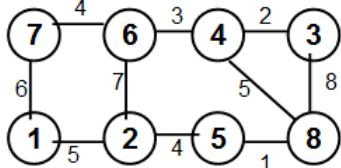
Atkins, Nayaa



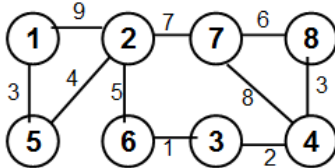
Barnett, Isaiah



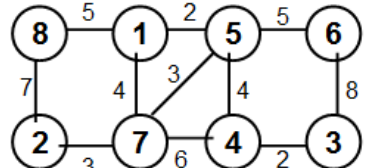
Dent, Kaitlyn



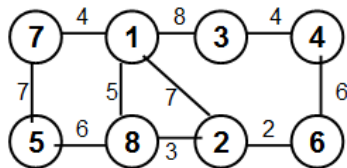
Drake, Keilah



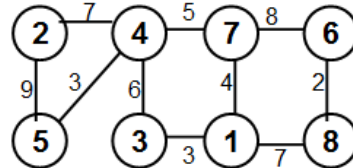
Harris, Chawne



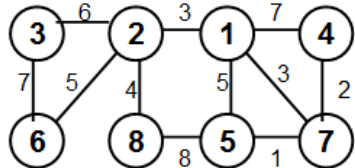
McGee, Bria



Redmond, Brandon



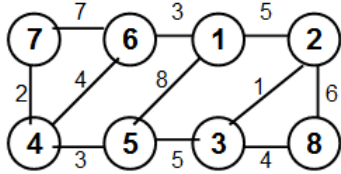
Roberts, Cambria



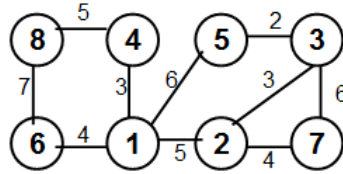
Student Name: \_\_\_\_\_

J#: \_\_\_\_\_

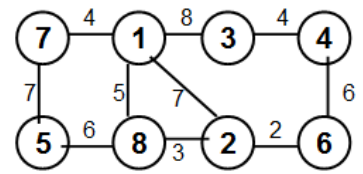
Stubbs, Jasmine



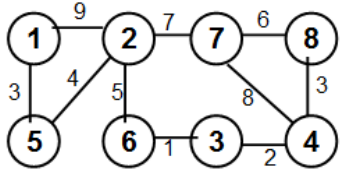
Swami, Shaurya



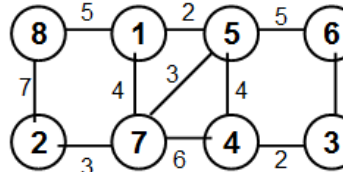
Tchakoua, Landrie



Teshome, Nahom



Triplett, Marzell



Wilkes, Kayla

