MODULE SIX PROBLEM SET

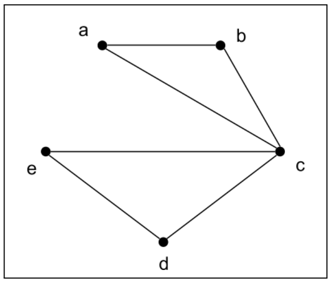
Ryan Hatch

Directions: Type your solutions into this document and be sure to show all steps for arriving at your solution. Just giving a final number may not receive full credit.

Problem 1

For parts (a) and (b), indicate if each of the two graphs are equal. Justify your answer.

(a)



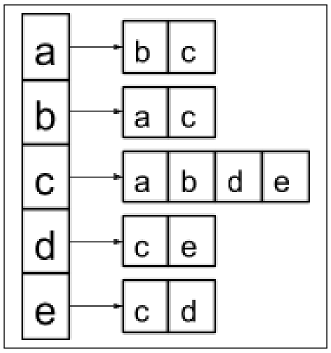


Figure 1: Left: An undirected graph has 5 vertices. The vertices are arranged in the form of an inverted pentagon. From the top left vertex, moving clockwise, the vertices are labeled: a, b, c, d, and e. Undirected edges, line segments, are between the following vertices: a and b; a and c; b and c; c and d; e and d; and e and c.

Figure 2: Right: The adjacency list representation of a graph. The list shows all the vertices, a through e, in a column from top to bottom. The adjacent vertices for each vertex in the column are placed in a row to the right of the corresponding vertexs cell in the column. An arrow points from each cell in the column to its corresponding row on the right. Data from the list, as follows: Vertex a is adjacent to vertices b and c. Vertex b is adjacent to vertices a and c. Vertex c is adjacent to vertices a, b, d, and e. Vertex d is adjacent to vertices c and e. Vertex e is adjacent to vertices c and d.

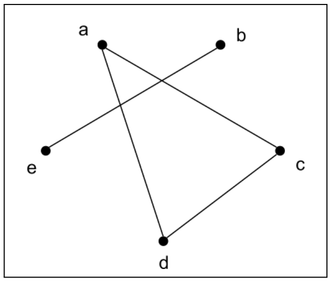
In order to see if the two graphs are equal, I had to make sure that they have the same set of vertices and the same set of edges.

Graph in Figure 1:
Vertices: a, b, c, d, e
Edges: ab, ac, bc, cd, ed, ec

Graph (adjacency list) in Figure 2:
Vertex a: connected to b, c
Vertex b: connected to a, c
Vertex c: connected to a, b, d, e
Vertex d: connected to c, e
Vertex e: connected to c, d

The adjacency list for Figure 2 matches the connections in the graph from Figure 1, so these two representations are of the same graph. Thus, the graphs in Figure 1 and Figure 2 are equal.

(b)



$$\begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Figure 3: *An undirected graph has 5 vertices. The vertices are arranged in the form of an inverted pentagon. Moving clockwise from the top left vertex a, the other vertices are, b, c, d, and e. Undirected edges, line segments, are between the following vertices: a and c; a and d; d and c; and e and b.*

To see if adjacency matrix corresponds to the graph in Figure 3.

Vertices: a, b, c, d, e
Edges: ac, ad, dc, eb
Adjacency Matrix:
a b c d e
a 0 0 1 1 0
b 0 0 0 0 1
c 1 0 0 1 0
d 1 0 1 0 1
e 0 1 0 1 0

The adjacency matrix shows the following connections:
Vertex a is connected to c, d
Vertex b is connected to e
Vertex c is connected to a, d
Vertex d is connected to a, c, e
Vertex e is connected to b, d

These connections match the edges listed in the visual representation of the graph in Figure 3. Thus, the graph and its adjacency matrix representation are equal.

(c) Prove that the two graphs below are isomorphic.

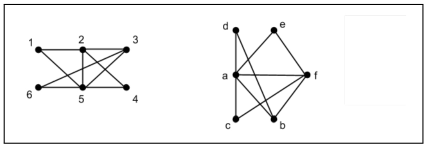


Figure 4: *Two undirected graphs. Each graph has 6 vertices. The vertices in the first graph are arranged in two rows and 3 columns. From left to right, the vertices in the top row are 1, 2, and 3. From left to right, the vertices in the bottom row are 6, 5, and 4. Undirected edges, line segments, are between the following vertices: 1 and 2; 2 and 3; 1 and 5; 2 and 5; 5 and 3; 2 and 4; 3 and 6; 6 and 5; and 5 and 4. The vertices in the second graph are a through f. Vertices d, a, and c, are vertically inline. Vertices e, f, and b, are horizontally to the right of vertices d, a, and c, respectively. Undirected edges, line segments, are between the following vertices: a and d; a and c; a and e; a and b; d and b; a and f; e and f; c and f; and b and f.*

For two graphs to be isomorphic, there must be a one-to-one correspondence between their vertices such that the number of edges and the connectivity is preserved.

Graph in the first diagram of Figure 4:
Vertices: 1, 2, 3, 4, 5, 6
Edges: 12, 23, 15, 25, 53, 24, 36, 65, 54

Graph in the second diagram of Figure 4:
Vertices: a, b, c, d, e, f
Edges: ad, ac, ae, ab, db, af, ef, cf, bf

To prove isomorphism, I had to find a mapping from the vertices of the first graph to the second such that the edges correspond.

For example:
Map 1 to a, 2 to d, 3 to c, 4 to f, 5 to e, and 6 to b.
If the edges are checked, you will see that each edge in the first graph has a corresponding edge in the second graph with the mapped vertices.
Thus, the two graphs are isomorphic because there is a one-to-one correspondence between their vertices that preserves the edges.

(d) Show that the pair of graphs are not isomorphic by showing that there is a property that is preserved under isomorphism which one graph has and the other does not.

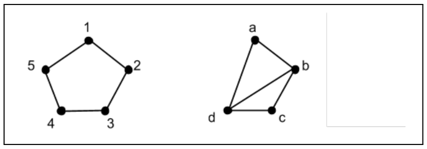


Figure 5: *Two undirected graphs. The first graph has 5 vertices, in the form of a regular pentagon. From the top vertex, moving clockwise, the vertices are labeled: 1, 2, 3, 4, and 5. Undirected edges, line segments, are between the following vertices: 1 and 2; 2 and 3; 3 and 4; 4 and 5; and 5 and 1. The second graph has 4 vertices, a through d. Vertices d and c are horizontally inline, where vertex d is to the left of vertex c. Vertex a is above and between vertices d and c. vertex b is to the right and below vertex a, but above the other two vertices. Undirected edges, line segments, are between the following vertices: a and b; b and c; a and d; d and c; d and b.*

To show two graphs are not isomorphic, I had to identify a graph invariant (a property that does not change under isomorphism) that is different in the two graphs.

In this case, I used the degree sequence as an invariant.
The degree sequence for the first graph is (2, 2, 2, 2, 2) since every vertex has exactly two neighbors.

For the second graph, the degree sequence is (3, 2, 2, 1), since there is one vertex connected to three others, two vertices connected to two others, and one vertex connected to only one other.

Since the degree sequences are not the same, the two graphs cannot be isomorphic. Another invariant is the number of cycles: the first graph contains a single cycle that includes all its vertices, whereas the second graph does not have a cycle that includes all its vertices. This also shows that the graphs are not isomorphic.

Problem 2

Refer to the undirected graph provided below:

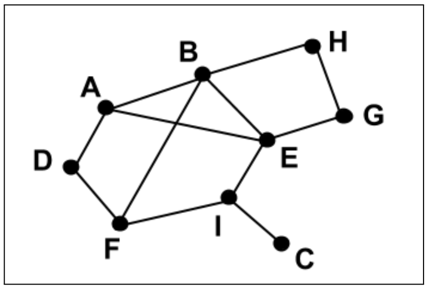


Figure 6: *An undirected graph has 9 vertices. 6 vertices form a hexagon, which is tilted upward to the right. Starting from the leftmost vertex, moving clockwise, the vertices forming the hexagon shape are: D, A, B, E, I, and F. Vertex H is above and to the right of vertex B. Vertex G is the rightmost vertex, below vertex H and above vertex E. Vertex C is the bottom most vertex, a little to the right of vertex E. Undirected edges, line segments, are between the following vertices: A and D; A and B; B and F; B and H; H and G; G and E; B and E; A and E; E and I; I and C; I and F; and F and D.*

(i) What is the maximum length of a path in the graph? Give an example of a path of that length.

The maximum length of a path is 8.
An example of such a path is C - I - E - G - H - B - A - D - F.

(ii) What is the maximum length of a cycle in the graph? Give an example of a cycle of that length.

The maximum length of a cycle is 9.
An example of such a cycle is A - D - F - I - C - I - E - G - H - B - A.

(iii) Give an example of an open walk of length five in the graph that is a trail but not a path.

An example of such a trail is D - A - B - H - B - E.

(iv) Give an example of a closed walk of length four in the graph that is not a circuit.

An example of this kind of walk is H - B - E - G - H.

(v) Give an example of a circuit of length zero in the graph.

An example of such a circuit is vertex A by itself.

Problem 3

(a) Find the connected components of each graph.

(i) $G = (V, E)$. $V = \{a, b, c, d, e\}$. $E = \emptyset$

The connected components are: $\{a\}$, $\{b\}$, $\{c\}$, $\{d\}$, $\{e\}$.

(ii) $G = (V, E)$. $V = \{a, b, c, d, e, f\}$. $E = \{\{c, f\}, \{a, b\}, \{d, a\}, \{e, c\}, \{b, f\}\}$

The connected component is the entire graph: $\{a, b, c, d, e, f\}$.

(b) Determine the edge connectivity and the vertex connectivity of each graph.

(i)

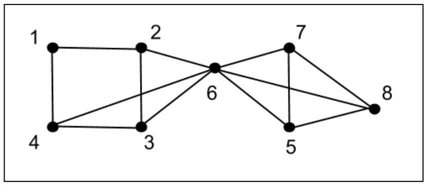


Figure 7: *An undirected graph has 8 vertices, 1 through 8. 4 vertices form a rectangular-shape on the left. Starting from the top left vertex and moving clockwise, the vertices of the rectangular shape are, 1, 2, 3, and 4. 3 vertices form a triangle on the right, with a vertical side on the left and the other vertex on the extreme right. Starting from the top vertex and moving clockwise, the vertices of the triangular shape are, 7, 8, and 5. Vertex 6 is between the rectangular shape and the triangular shape. Undirected edges, line segments, are between the following vertices: 1 and 2; 2 and 3; 3 and 4; 4 and 1; 2 and 6; 4 and 6; 3 and 6; 6 and 7; 6 and 8; 6 and 5; 7 and 5; 7 and 8; and 5 and 8.*

The edge connectivity $\kappa'(G) = 2$ and the vertex connectivity $\kappa(G) = 1$.

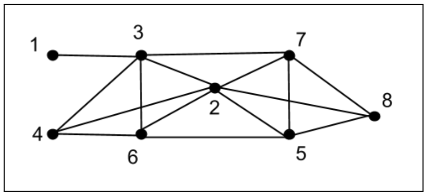
(ii) 

Figure 8: *An undirected graph has 8 vertices, 1 through 8. 4 vertices form a rectangular shape in the center. Starting from the top left vertex and moving clockwise, the vertices of the rectangular shape are, 3, 7, 5, and 6. Vertex 2 is at about the center of the rectangular shape. Vertex 8 is to the right of the rectangular shape. Vertex 1 and 4 are to the left of the rectangular shape, horizontally in-line with vertices 3 and 6, respectively. Undirected edges, line segments, are between the following vertices: 1 and 3; 3 and 7; 3 and 4; 3 and 6; 3 and 2; 4 and 2; 4 and 6; 6 and 2; 6 and 5; 2 and 5; 2 and 7; 2 and 8; 7 and 5; 7 and 8; and 5 and 8.*

The edge connectivity $\kappa'(G) = 3$ and the vertex connectivity $\kappa(G) = 1$.

PROBLEM 4

For parts (a) and (b) below, find an Euler circuit in the graph or explain why the graph does not have an Euler circuit.

(a)

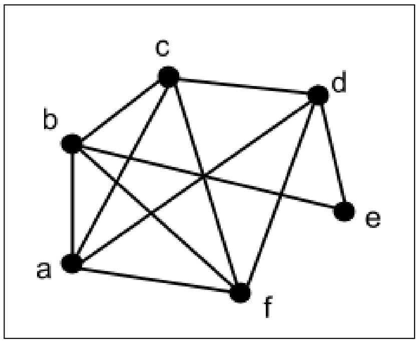


Figure 9: *An undirected graph has 6 vertices, a through f. 5 vertices are in the form of a regular pentagon, rotated 90 degrees clockwise. Hence, the top vertex becomes the rightmost vertex. From the bottom left vertex, moving clockwise, the vertices in the pentagon shape are labeled: a, b, c, e, and f. Vertex d is above vertex e, below and to the right of vertex c. Undirected edges, line segments, are between the following vertices: a and b; a and c; a and d; a and f; b and f; b and c; b and e; c and d; d and e; and d and f. Edges c f, a d, and b e intersect at the same point.*

There is no Euler circuit in the graph of Figure 9 because all vertices have an odd degree. Therefore, the graph does not have an Euler circuit.

(b)

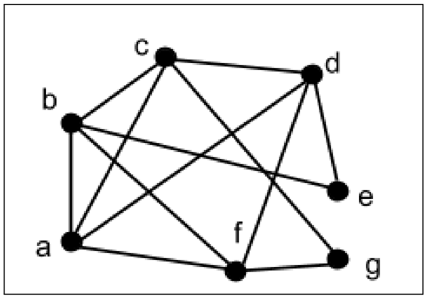


Figure 10: *An undirected graph has 7 vertices, a through g. 5 vertices are in the form of a regular pentagon, rotated 90 degrees clockwise. Hence, the top vertex becomes the rightmost vertex. From the bottom left vertex, moving clockwise, the vertices in the pentagon shape are labeled: a, b, c, e, and f. Vertex d is above vertex e, below and to the right of vertex c. Vertex g is below vertex e, above and to the right of vertex f. Undirected edges, line segments, are between the following vertices: a and b; a and c; a and d; a and f; b and f; b and c; b and e; c and d; c and g; d and e; d and f; and f and g.*

The Euler circuit for the graph is ‘a - b - e - c - d - a - f - g - c - b - f - d - e - g - f - a‘.
This circuit visits every edge exactly once and returns to the starting vertex, so it is an Euler circuit.

(c) For each graph below, find an Euler trail in the graph or explain why the graph does not have an Euler trail.

(Hint: One way to find an Euler trail is to add an edge between two vertices with odd degree, find an Euler circuit in the resulting graph, and then delete the added edge from the circuit.)

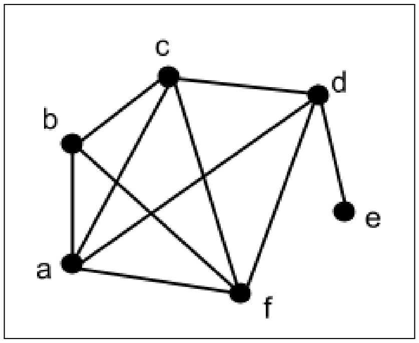
(i) 

Figure 11: *An undirected graph has 6 vertices, a through f. 5 vertices are in the form of a regular pentagon, rotated 90 degrees clockwise. Hence, the top vertex becomes the rightmost vertex. From the bottom left vertex, moving clockwise, the vertices in the pentagon shape are labeled: a, b, c, e, and f. Vertex d is above vertex e, below and to the right of vertex c. Undirected edges, line segments, are between the following vertices: a and b; a and c; a and d; a and f; b and f; b and c; c and d; c and f; d and e; and d and f.*

An Euler trail can be created by adding an edge between two vertices with odd degrees and finding an Euler circuit in the modified graph. Then, remove the added edge to have an Euler trail.

The vertices with odd degrees are a and e. Adding an edge between a and e creates a graph with all even degrees.

An Euler circuit in the modified graph is 'a - b - c - d - e - f - a - d - c - f - b - a'.
Removing the added edge between a and e from the circuit results in an Euler trail: 'a - b - c - d - e - f - a - d - c - f - b'.

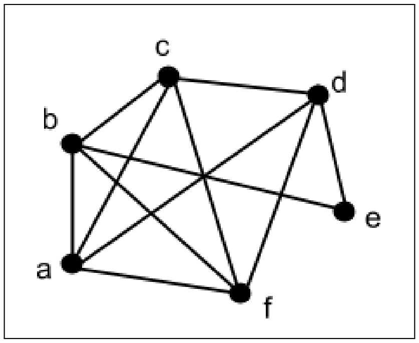
(ii) 

Figure 12: *An undirected graph has 6 vertices, a through f. 5 vertices are in the form of a regular pentagon, rotated 90 degrees clockwise. Hence, the top vertex becomes the rightmost vertex. From the bottom left vertex, moving clockwise, the vertices in the pentagon shape are labeled: a, b, c, e, and f. Vertex d is above vertex e, below and to the right of vertex c. Undirected edges, line segments, are between the following vertices: a and b; a and c; a and d; a and f; b and f; b and c; b and e; c and d; d and e; and d and f. Edges c f, a d, and b e intersect at the same point.*

The same method used for Figure 11 can also be applied to the graph in Figure 12 to find an Euler trail. The vertices with odd degrees are a and e. Adding an edge between a and e creates a graph with all even degrees.
An Euler circuit in the modified graph is 'a - b - c - d - e - f - a - d - c - f - b - a'.
Removing the added edge between a and e from the circuit results in an Euler trail: 'a - b - c - d - e - f - a - d - c - f - b'.

Problem 5

Consider the following tree for a prefix code:

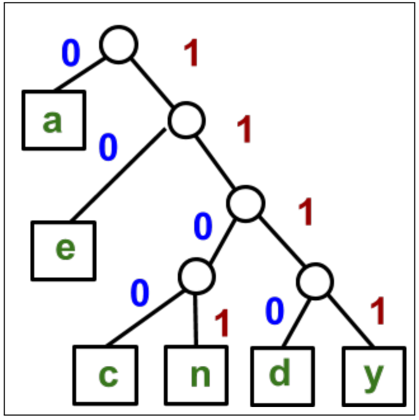


Figure 13: *A tree with 5 vertices. The top vertex branches into character, a, on the left, and a vertex on the right. The vertex in the second level branches into character, e, on the left, and a vertex on the right. The vertex in the third level branches into two vertices. The left vertex in the fourth level branches into character, c, on the left, and character, n, on the right. The right vertex in the fourth level branches into character, d, on the left, and character, y, on the right. The weight of each edge branching left from a vertex is 0. The weight of each edge branching right from a vertex is 1.*

(a) Use the tree to encode “day”.

To encode “day” using the tree, I got the following binary sequences for each character:

- d: 11
- a: 0
- y: 111

Thus, ”day” is encoded as 011111.

(b) Use the tree to encode “candy”.

To encode “candy” using the tree, I got the following binary sequences for each character:

- c: 100

- a: 0
- n: 101
- d: 11
- y: 111

Thus, ”candy” is encoded as 010010111111.

(c) Use the tree to decode “1110101101″.

To decode “1110101101” using the tree, I followed the binary sequence to get the characters:

- 111: y
- 0: a
- 101: n
- 10: d

The sequence decodes to ”yand”.

(d) Use the tree to decode “111001101110010″.

To decode “111001101110010” using the tree, I once again followed the binary sequence to get the characters:

- 111: y
- 0: a
- 11: d
- 0: a
- 11: d
- 101: n
- 0: a
- 10: d

The sequence decodes to ”yadadna”.

Problem 6

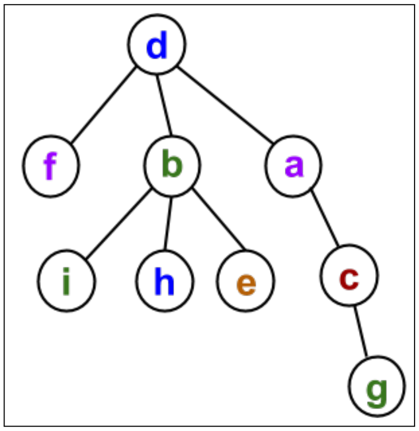


Figure 14: *A tree diagram has 9 vertices. The top vertex is d. Vertex d has three branches to vertices, f, b, and a. Vertex b branches to three vertices, i, h, and e. Vertex a branches to vertex c. Vertex c branches to vertex g.*

(a) Give the order in which the vertices of the tree are visited in a post-order traversal.

The order in which the vertices of the tree are visited in a post-order traversal is: f, i, h, e, b, g, c, a, d.

(b) Give the order in which the vertices of the tree are visited in a pre-order traversal.

The order in which the vertices of the tree are visited in a pre-order traversal is: d, f, b, i, h, e, a, c, g.

Problem 7

Consider the following tree. Assume that the neighbors of a vertex are considered in alphabetical order.

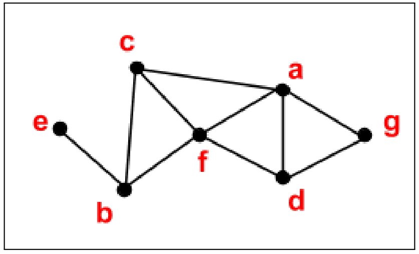


Figure 15: *A graph has 7 vertices, a through g, and 10 edges. Vertex e on the left end is horizontally inline with vertex g on the right end. Vertex b is below and to the right of vertex e. Vertex c is above vertex e and to the right of vertex b. Vertex f is between and to the right of vertices c and b. Vertex f is horizontally inline with vertices e and g. Vertex a is above and to the right of vertex f. Vertex d is below and to the right of vertex f. Vertex a is vertically inline with vertex d. Vertex g is between and to the right of vertices a and d. The edges between the vertices are as follows: e and b; b and c; c and f; c and a; a and d; b and f; f and a; f and d; a and g; and d and g.*

(a) Give the tree resulting from a traversal of the graph below starting at vertex a using BFS.

The tree resulting from a traversal of the graph below starting at vertex a using BFS is:
a – c – b – e, a – d, a – f, a – g.

(b) Give the tree resulting from a traversal of the graph below starting at vertex a using DFS.

The tree resulting from a traversal of the graph below starting at vertex a using DFS is:
a – c – b – e, c – f – d, a – g.

Problem 8

An undirected weighted graph G is given below:

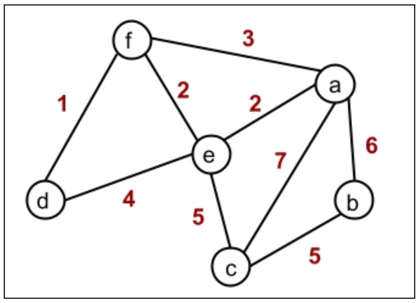


Figure 16: *An undirected weighted graph has 6 vertices, a through f, and 9 edges. Vertex d is on the left. Vertex f is above and to the right of vertex d. Vertex e is below and to the right of vertex f, but above vertex d. Vertex c is below and to the right of vertex e. Vertex a is above vertex e and to the right of vertex c. Vertex b is below and to the right of vertex a, but above vertex c. The edges between the vertices and their weight are as follows: d and f, 1; d and e, 4; f and e, 2; e and a, 2; f and a, 3; e and c, 5; c and a, 7; c and b, 5; and a and b, 6.*

(a) Use Prim's algorithm to compute the minimum spanning tree for the weighted graph. Start the algorithm at vertex a. Show the order in which the edges are added to the tree.

Using Prim's algorithm starting at vertex a, the minimum spanning tree is constructed by adding the edges in the following order:

- Add edge a, e with weight 2.
- Add edge e, f with weight 2.
- Add edge d, f with weight 1.
- Add edge c, a with weight 5.
- Add edge c, b with weight 5.

The edges added form the minimum spanning tree with a total weight of 15.

(b) What is the minimum weight spanning tree for the weighted graph in the previous question subject to the condition that edge $\{d, e\}$ is in the spanning tree?

To include the edge d, e in the minimum spanning tree, the edges are added in the following order:

- Include edge d, e by default.
- Add edge d, f with weight 1.
- Add edge e, a with weight 2.
- Add edge e, f with weight 2.
- Add edge c, b with weight 5.

The edges added form the minimum spanning tree with the edge d, e included and a total weight of 17.

(c) How would you generalize this idea? Suppose you are given a graph G and a particular edge $\{u, v\}$ in the graph. How would you alter Prim's algorithm to find the minimum spanning tree subject to the condition that $\{u, v\}$ is in the tree?

To generalize this idea for any given edge u, v in the graph, Prim's algorithm is modified as follows:

- Include the edge u, v in the tree from the beginning.
- Perform Prim's algorithm starting from vertex u or v, considering all other edges except u, v.
- Grow the tree by selecting the edges with the smallest weight while avoiding cycles until all vertices are included in the tree.

This makes sure that the resulting tree is a minimum spanning tree with the specific edge u, v included.