Graph Theory

Homework 2

1. Define a tree and a leaf.

2. Given a graph G, prove that the statements “G is connected and has no cycles” and “G has n−1 edges and no cycles” are equivalent.

3. Choose the correct statement.
   A) Every graph contains a spanning tree.
   B) A tree has *n-1* cut-edges.
   C) The radius of a tree is *n-1*.
   D) The diameter of a tree is *n-1*.

   (*n* is the number of vertices in the tree)

4. Let T be a tree. It is known that T has
   a. One vertex of degree 4
   b. Two vertices of degree 3
   c. three vertices of degree 2
   d. All remaining vertices are leaves.

   What is **the total number of vertices** in the tree?

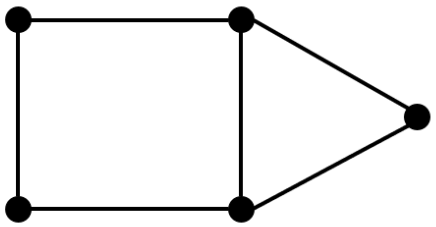

Figure 1.

5. Based on Figure 1, how many spanning trees are there in the graph?

6. Based on Figure 1, what is the Wiener index of the graph? (Assuming that the weight of all edges is 1)

7. How many spanning trees are there in the graph K_5?

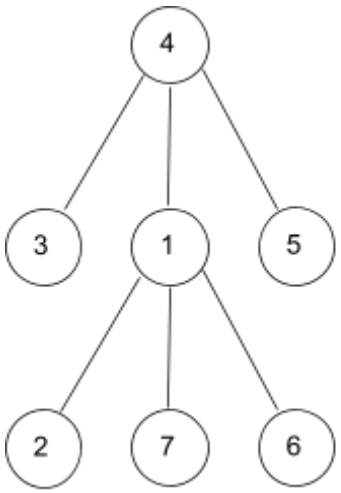

Figure. 2

8. Based on Figure 2, write the Prüfer code of the tree.

9. According to the answer to question 8, show the process of rebuilding the tree.

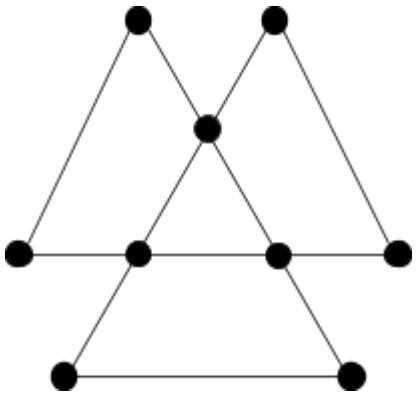

Figure. 3

10. Find the eccentricity of each vertex of the graph in Figure 3.

11. Based on the above question, what are the radius, the diameter, and the length of the longest path of Figure 3?

12. According to the slides, describe what $\tau(G)$ represents.

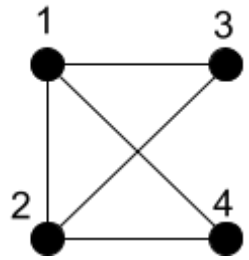

Figure. 4

13. Based on Figure 4, use the Matrix Tree Theorem to find the number of spanning trees of the graph.

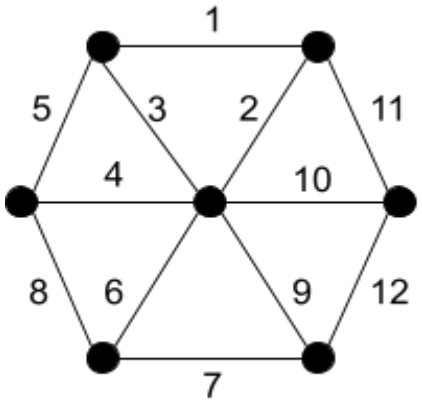

Figure. 5

14. Use Kruskal's algorithm to find the minimum spanning tree of Figure 5.

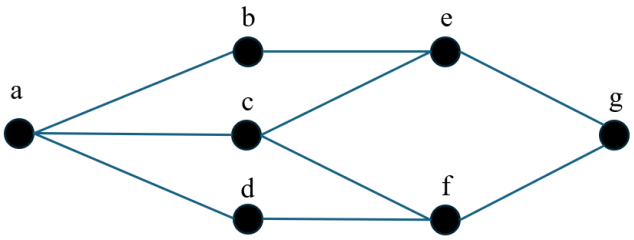

Figure. 6

15. Perform BFS starting from node a, and list the order in which the nodes are visited. When there are multiple neighbors at the same level, visit them in alphabetical order.(Figure 6)

16. Starting from **node b**, use BFS to find the **shortest path** to all other nodes. List the distance from b to each node.(Figure 6)

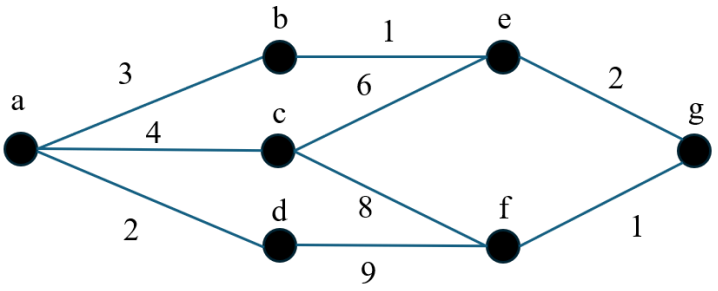

Figure 7.

17. Why might the result be incorrect when using Dijkstra's algorithm to calculate the shortest path in a graph that contains negative edge weights? Please provide an example.

18. Use **Dijkstra's algorithm** to calculate the shortest path from **node a** to all other nodes. Please list the **shortest distance to each node.**(Figure 7)

19. In the Chinese Postman Problem, if a graph has an **even degree** for all vertices, what is the minimum number of times the postman needs to traverse each edge?
    a. 1
    b. 2
    c. 3
    d. 4

20. In the Chinese Postman Problem, if a graph has **two odd-degree vertices**, how should the postman proceed to minimize the distance traveled?
    a. Traverse all edges once and return to the starting point.
    b. Pair up the two odd-degree vertices, traverse the shortest path between them twice, and then complete the rest of the graph.
    c. Start from one of the odd-degree vertices and traverse the edges randomly.
    d. Start from any vertex and follow the edges in any direction.