Graph Theory Problem Set

Topics: Adjacency Lists, Adjacency Matrices, DFS, BFS, Dijkstra's Algorithm, and Minimum Spanning Trees

Section A: Multiple Choice Questions

Q1) Select all statements that are TRUE about adjacency matrix representation:

A) The space complexity for an adjacency matrix of a graph with V vertices is $O(V^2)$
B) Checking if an edge exists between two vertices takes O(1) time
C) Adding a new vertex to the graph takes O(1) time
D) For sparse graphs, adjacency matrix is more space-efficient than adjacency list
E) The adjacency matrix of an undirected graph is always symmetric

Q2) What is the time complexity of BFS traversal on a graph with V vertices and E edges represented using an adjacency list?

a) O(V)
b) O(E)
c) O(V + E)
d) O(V × E)

Q3) For the adjacency matrix of an undirected graph, what can be said about the diagonal elements?

a) They are always 0 unless there are self-loops
b) They are always 1
c) They represent the degree of each vertex
d) They are undefined

Q4) In DFS traversal of a graph, which data structure is implicitly used?

a) Queue
b) Stack
c) Heap
d) Hash Table

Q5) Which of the following statements about Dijkstra's algorithm are TRUE?

A) It works correctly on graphs with negative edge weights
B) It finds the shortest path from a source vertex to all other vertices
C) It uses a priority queue/min-heap for efficient implementation
D) Its time complexity with a binary heap is O(E log V)
E) It can detect negative cycles

Q6) What is the maximum number of edges in a complete undirected graph with n vertices?

a) n
b) n - 1
c) n(n-1)/2
d) n(n-1)

Q7) Which algorithm(s) can be used to find a Minimum Spanning Tree?

A) Kruskal's Algorithm
B) Prim's Algorithm
C) Dijkstra's Algorithm
D) Floyd-Warshall Algorithm
E) Bellman-Ford Algorithm

Q8) In BFS traversal, nodes are visited in what order relative to the source?

a) Increasing order of their degree
b) Decreasing order of their degree
c) Increasing order of their distance from source
d) Random order

Q9) A Minimum Spanning Tree of a graph with V vertices must have how many edges?

a) V
b) V - 1
c) V + 1
d) 2V

Q10) Which property must a graph satisfy for Kruskal's algorithm to work?

a) Graph must be directed
b) Graph must be connected
c) Graph must have positive weights
d) All edges must have unique weights

Section B: Short Answer Questions

Q11) Consider the following undirected graph:

Graph Representation: Vertices: {A, B, C, D, E} Edges with weights: A -- B (weight: 4) A -- C (weight: 2) B -- C (weight: 1) B -- D (weight: 5) C -- D (weight: 8) C -- E (weight: 10) D -- E (weight: 2)

a) Represent this graph as an adjacency matrix.
b) Represent this graph as an adjacency list.
c) What would be different if the graph were directed from the first vertex to the second in each edge?

Q12) Given the adjacency matrix below:

```
  A B C D
A [0 1 1 0]
B [1 0 0 1]
C [1 0 0 1]
D [0 1 1 0]
```

a) Draw the graph represented by this matrix.
b) Is this graph directed or undirected? How can you tell?
c) What is the degree of each vertex?

Q13) Perform BFS and DFS traversals on the following graph starting from vertex A:

Graph Structure: Vertices: {A, B, C, D, E, F} Edges: A-B, A-C, B-D, B-E, C-F, E-F Adjacency Lists (alphabetically sorted): A: [B, C] B: [A, D, E] C: [A, F] D: [B] E: [B, F] F: [C, E]

a) Write the order in which vertices are visited using BFS starting from A.
b) Write the order in which vertices are visited using DFS starting from A.

Q14) Consider the following weighted graph for Dijkstra's algorithm:

Vertices: {S, A, B, C, D, T} Edges (directed) with weights: S   A (weight: 2) S   B (weight: 4) A   B (weight: 1) A   C (weight: 7) B   C (weight: 3) B   D (weight: 5) C   T (weight: 2) D   T (weight: 1)

Starting from vertex S, apply Dijkstra's algorithm to find the shortest path to all vertices. Show:
a) The order in which vertices are finalized (removed from priority queue)
b) The shortest distance to each vertex from S
c) The shortest path from S to T

Q15) Apply Kruskal's algorithm to find the Minimum Spanning Tree:

Vertices: {A, B, C, D, E, F} Edges with weights: A-B: 3 A-C: 5 B-C: 2 B-D: 4 C-D: 6 C-E: 8 D-E: 3 D-F: 7 E-F: 4

a) List the edges in the order they are considered by Kruskal's algorithm.
b) Which edges are included in the MST?
c) What is the total weight of the MST?

Section C: Long Answer Questions

Q16) Answer True or False with brief justification:

1. In an undirected graph, the sum of degrees of all vertices is always even.

2. BFS can be used to find the shortest path in a weighted graph.

3. If all edges in a graph have distinct weights, then the MST is unique.

4. DFS can be used to detect cycles in an undirected graph.

5. Dijkstra's algorithm works correctly on graphs with negative edge weights.

6. A graph with n vertices and n-1 edges is always a tree.

7. The time complexity of Prim's algorithm with a binary heap is O(E log V).

8. BFS uses less memory than DFS for traversing a graph.

9. Every connected graph has at least one spanning tree.

10. In an adjacency list representation, checking if an edge exists takes O(1) time.

Q17) Programming Problem - Connected Components:

You are given an undirected graph with n vertices (0 to n-1) and a list of edges. Some vertices may not be connected to any other vertices.

Task: Write an algorithm to find the number of connected components in the graph. Explain your approach and analyze its time complexity.

Example:
Input: n = 6, edges = [(0,1), (1,2), (3,4)]
Output: 3
Explanation: Components are {0,1,2}, {3,4}, and {5}

Q18) Programming Problem - Path Existence:

Given a directed graph and two vertices source and destination, determine if there exists a path from source to destination.

Task: Describe an algorithm to solve this problem. Which graph traversal algorithm would you use and why? What is the time complexity?

Q19) Comparison Question:

Compare and contrast the following:

a) **Adjacency Matrix vs Adjacency List:**
Discuss space complexity, time complexity for common operations (add edge, remove edge, check if edge exists, find all neighbors), and when each representation is preferred.

b) **BFS vs DFS:**
Discuss their traversal order, data structures used, applications, and when one is preferred over the other.

Q20) Apply Prim's algorithm to find the MST for the following graph, starting from vertex A:

Vertices: {A, B, C, D, E} Edges with weights: A-B: 7 A-D: 5 B-C: 8 B-D: 9 B-E: 7 C-E: 5 D-E: 15 D-F: 6 E-F: 8 E-G: 9 F-G: 11 (where F and G are additional vertices)

Show step-by-step:
a) Which edges are added to the MST at each step
b) The state of the priority queue after each step
c) The final MST and its total weight

Note: For all graph problems, clearly show your work and explain your reasoning. For algorithms, indicate the order of vertex/edge processing.

End of Problem Set