

## Lecture 13- Shortest Paths

**1. Which algorithm is optimal for finding the shortest path in an unweighted graph?**

- A) Kruskal's algorithm
- B) DFS (Depth-First Search)
- C) BFS (Breadth-First Search)
- D) Topological Sort

**Answer:** C) BFS (Breadth-First Search)

---

**2. What data structures are used in BFS to track the shortest distance (SD) and previous node (PN)?**

- A) Stack and Queue
- B) Priority Queue and Hash Map
- C) Two maps: one for SD and one for PN
- D) Adjacency List and Array

**Answer:** C) Two maps: one for SD and one for PN

---

**3. Dijkstra's Algorithm is NOT suitable for graphs with:**

- A) Non-negative edge weights
- B) Directed edges
- C) Negative edge weights
- D) Undirected edges

**Answer:** C) Negative edge weights

---

**4. What is the time complexity of Dijkstra's Algorithm using a binary min-heap?**

- A)  $O(V + E)$
- B)  $O((V + E) \log V)$
- C)  $O(V^2)$
- D)  $O(E \log V)$

**Answer:** B)  $O((V + E) \log V)$

---

**5. During edge relaxation, if a shorter path to node  $v$  is found through node  $u$ , what happens?**

- A)  $SD[v]$  and  $PN[v]$  are updated
- B)  $SD[u]$  is reset to infinity
- C)  $PN[u]$  is set to  $v$
- D) The edge  $u \rightarrow v$  is removed

**Answer:** A)  $SD[v]$  and  $PN[v]$  are updated

---

**6. In Dijkstra's algorithm, how is the priority queue used to select the next node to visit?**

- A) The node with the largest known distance is chosen first.
- B) The node with the smallest Shortest Distance (SD) is chosen first.
- C) Nodes are processed in alphabetical order.
- D) Nodes are selected based on the number of edges.

**Answer:** B) The node with the smallest Shortest Distance (SD) is chosen first.

---

**7. In the Shortest Path Tree (SPT), what does the PN map store?**

- A) The shortest distance from the source
- B) The previous node in the shortest path
- C) The weight of edges
- D) The next node to visit

**Answer:** B) The previous node in the shortest path

---

**8. What is the first step in Dijkstra's Algorithm?**

- A) Relax all edges
- B) Initialize SD for the source node to 0 and others to  $\infty$
- C) Mark all nodes as known
- D) Sort edges by weight

**Answer:** B) Initialize SD for the source node to 0 and others to  $\infty$

---

**9. Which algorithm processes nodes in topological order for shortest paths?**

- A) BFS
- B) Dijkstra's Algorithm
- C) Bellman-Ford Algorithm
- D) Topological Sort-based algorithm

**Answer:** D) Topological Sort-based algorithm

---

**10. In BFS, nodes are visited in order of:**

- A) Increasing edge weights
- B) Decreasing distance from the source
- C) Increasing distance from the source
- D) Alphabetical order

**Answer:** C) Increasing distance from the source

---

**11. After running Dijkstra's Algorithm, how is the shortest path reconstructed?**

- A) Using the adjacency list
- B) Following backpointers in the PN map
- C) Sorting the SD map

D) Reversing the visit order

**Answer:** B) Following backpointers in the PN map

---

**12. What is a key advantage of using Topological Sort for shortest paths in a DAG?**

A) Handles negative weights

B) Runs in linear time

C) Works for cyclic graphs

D) Uses a priority queue

**Answer:** B) Runs in linear time

---

**13. During the edge relaxation step in Dijkstra's algorithm, which condition must be met to update the shortest distance (SD) to node  $v$ ?**

A)  $SD[v] < SD[u] + w(u,v)$

B)  $SD[v] > SD[u] + w(u,v)$

C)  $SD[u] < SD[v] + w(u,v)$

D)  $SD[v] == SD[u] + w(u,v)$

**Answer:** B)  $SD[v] > SD[u] + w(u,v)$

---