

****BFS and DFS Quiz (Hard - Short Answer)****

****Instructions:** Answer the following questions concisely and accurately.**

1. ****Question:**** Describe a scenario where a Breadth-First Search (BFS) algorithm would be significantly more efficient than a Depth-First Search (DFS) algorithm.

****Answer:**** Finding the shortest path in an unweighted graph.

2. ****Question:**** What data structure is typically used to implement a BFS algorithm?

****Answer:**** Queue

3. ****Question:**** What data structure is typically used to implement a DFS algorithm?

****Answer:**** Stack or Recursion

4. ****Question:**** Explain the concept of "cycle detection" and how it differs in BFS and DFS implementations.

****Answer:**** Cycle detection involves finding loops in a graph. BFS detects cycles by finding a node already visited in the current level. DFS detects cycles by finding a node already visited in the current branch (recursive stack).

5. ****Question:**** True or False: A DFS algorithm guarantees finding the shortest path in an unweighted graph.

****Answer:**** False

6. ****Question:**** True or False: A BFS algorithm guarantees finding the shortest path in an unweighted graph.

****Answer:**** True

7. ****Question:**** Describe a situation where DFS would be preferred over BFS.

****Answer:**** Finding a path in a very deep, sparsely branched graph; exploring all possible paths in a game tree.

8. ****Question:**** What is the time complexity of BFS in a graph represented as an adjacency list?

****Answer:**** $O(V + E)$, where V is the number of vertices and E is the number of edges.

9. ****Question:**** What is the space complexity of BFS in the worst case scenario?

****Answer:**** $O(V)$

10. ****Question:**** What is the time complexity of DFS in a graph represented as an adjacency list?

****Answer:**** $O(V + E)$

11. ****Question:**** What is the space complexity of DFS in the worst case scenario?

****Answer:**** $O(V)$ in case of recursion; $O(V)$ in case of iterative implementation using a stack.

12. ****Question:**** How can you modify BFS to find all paths between two nodes?

****Answer:**** Track the paths taken from the source node to each node visited.

13. ****Question:**** How can you modify DFS to find all paths between two nodes?

****Answer:**** Track the paths taken from the source node recursively and backtrack.

14. ****Question:**** Describe how to use BFS to determine if a graph is bipartite.

****Answer:**** Assign colors (e.g., 0 and 1) to nodes during traversal. If a neighbor has the same color, it's not bipartite.

15. ****Question:**** How can DFS be used to detect articulation points (cut vertices) in a graph?

****Answer:**** By tracking the discovery time and low-link values for each node.

16. ****Question:**** What is topological sorting, and which algorithm (BFS or DFS) is commonly used to perform it?

****Answer:**** Arranging nodes such that for every directed edge from node A to node B, A appears before B. DFS is commonly used.

17. ****Question:**** Explain the concept of "strongly connected components" in a directed graph, and which algorithm is generally used to find them?

****Answer:**** A set of nodes where there is a path between any two nodes in the set. Kosaraju's algorithm (uses DFS) is a common solution.

18. ****Question:**** How does the use of a priority queue change the behavior of a BFS algorithm?

****Answer:**** It transforms it into a Dijkstra's algorithm, finding the shortest path in a weighted graph.

19. ****Question:**** What is a potential disadvantage of using recursion for DFS in terms of memory usage?

****Answer:**** Stack overflow if the graph is very deep.

20. ****Question:**** Give an example of a real-world problem where BFS would be an appropriate algorithmic solution.

****Answer:**** Finding all people within a certain degree of separation from a given person on a social network. (Many other valid answers exist).