Breadth-First Search (BFS) - Revision Notes

1. What is BFS?

Breadth-First Search (BFS) is a graph traversal algorithm that explores neighbors level by level before moving deeper. It uses a queue data structure.

2. Key Characteristics

- Explores graph level by level (like waves).
- Uses a queue (FIFO).
- Works for both adjacency list and adjacency matrix representations.
- Time Complexity: $O(V + E)$, where V = vertices, E = edges.
- Space Complexity: $O(V)$ for visited array and queue.

3. Applications of BFS

- Shortest path in an unweighted graph (like finding the minimum steps in a maze).
- Checking bipartiteness of a graph.
- Finding connected components (with multiple BFS runs).
- Cycle detection in undirected graphs.
- Web crawling, peer-to-peer networks, social networking apps (friend suggestions).

4. BFS Pseudocode

```
BFS(start): Create a queue Q Mark start as visited Enqueue start into Q while Q is
not empty: node = Dequeue from Q Process node for each neighbor of node: if neighbor
is not visited: Mark neighbor as visited Enqueue neighbor
```

5. Important Leetcode Problems

- 200 - Number of Islands
- 207 - Course Schedule (using BFS for Topological Sort)
- 323 - Number of Connected Components in an Undirected Graph
- 785 - Is Graph Bipartite?
- 994 - Rotting Oranges