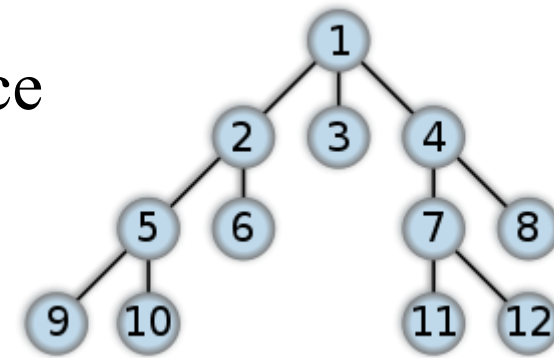# BFS and DFS
Problem Solving Club Nov 2 2016

Breadth First Search (BFS) Review

- What is the main problem that BFS solves?
- Single Source Shortest Path (SSSP) on an unweighted graph - find shortest paths from a source vertex to all other vertices in the graph
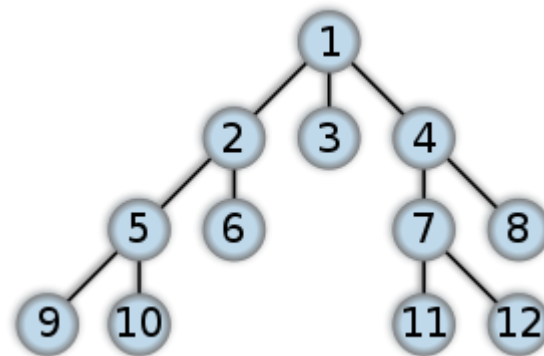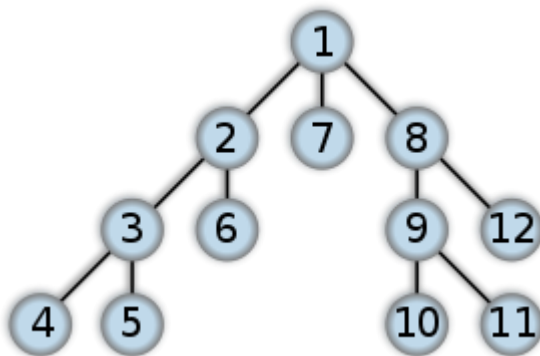

- What is the main data structure used in BFS?
- Queue

- What is the running time of BFS, in terms of V (number of vertices) and E (number of edges)?
- O(V+E)

Depth First Search (DFS)

- DFS is an algorithm closely related to BFS

- It is same algorithm, but the queue is replaced with a stack

- The only difference with BFS is the order in which vertices are visited

BFS vs DFS

- Why would we choose one over the other?

  – Finding shortest paths: BFS finds shortest paths. DFS does not.

  – Implementation: The stack in DFS can be implemented implicitly with the call stack. To some people, DFS is easier to implement than BFS.

```
bool vis[]
void dfs(n) {
  if vis[n]: return
  vis[n] = true
```

Floyd-Warshall algorithm

- Floyd-Warshall is an algorithm that solves the all-pairs shortest path problem in a weighted graph (even with negative weights!)

- It finds the shortest paths between every pair of vertices

- What makes Floyd-Warshall special is its extremely simple implementation

- for k from 1 to |V|
    - for i from 1 to |V|
        - for j from 1 to |V|
            - if dist[i][j] > dist[i][k] + dist[k][j]
                - dist[i][j] ← dist[i][k] + dist[k][j]

- What is the running time, in terms of V (number of vertices) and E (number of edges)?

Summary

Algorithm	Complexity	Description
Breadth-first search (BFS)	$O(V+E)$	Single-source shortest path on an unweighted graph.
Depth-first search (DFS)	$O(V+E)$	Take BFS and replace the queue with a stack. Does **not** find shortest paths. It can do many things BFS can do, like find connected components. It is commonly used as part of other algorithms and may be easier to implement than BFS.
Floyd-Warshall	$O(V^3)$	All-paths shortest paths on a weighted graph with negative weights. Very simple implementation. **Use whenever possible.**
Dijkstra's	$O((V+E)\log V)$ **or** $O(V^2)$	Take BFS and replace the queue with a priority queue. Single-source shortest path on a weighted graph with **no** negative weights. Complex implementation. Remember to consider Floyd-Warshall first.
Bellman-Ford	$O(VE)$	Single-source shortest path on a weighted graph with negative weights