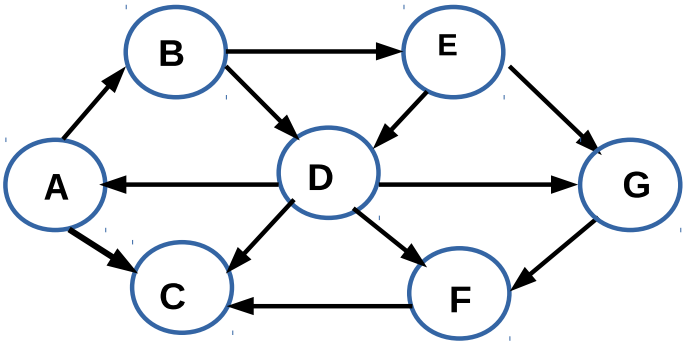
Breadth First Search - Example

- Consider the directed graph below, use the breadth first search procedure to give the BFS tree when A is the source node

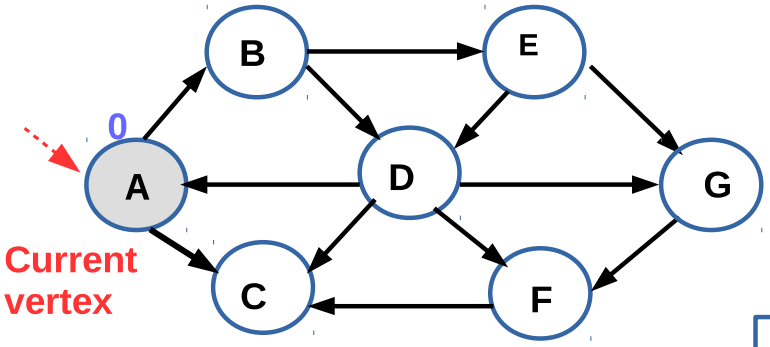


Source node

- **Note:** This is an instance of the unweighted shortest-path computation.
- We try to reach every other node in the graph from the source node and ensure each vertex is visited only once

Breadth First Search - Example

- Consider the directed graph below, use the breadth first search procedure to give the BFS tree when A is the source node



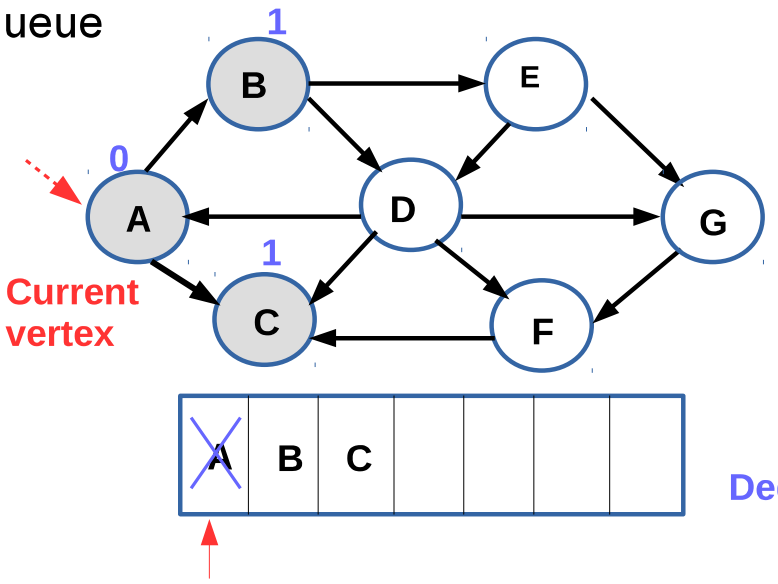
Source node

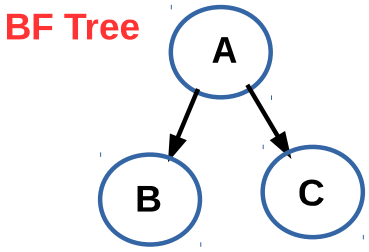
Queue

- The graph after the starting vertex has been marked as reachable in zero edges

Breadth First Search - Example

- Now check all adjacent vertices to the “current node” and enqueue



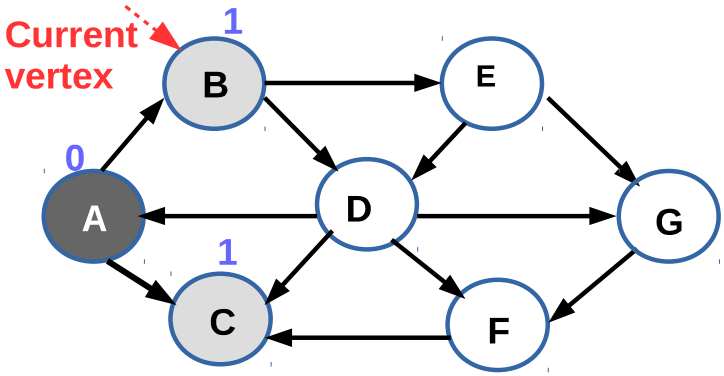


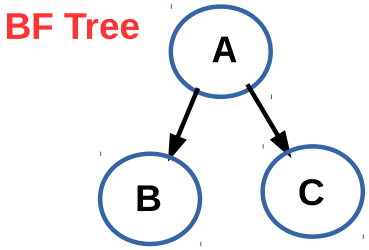
Dequeue current working vertex

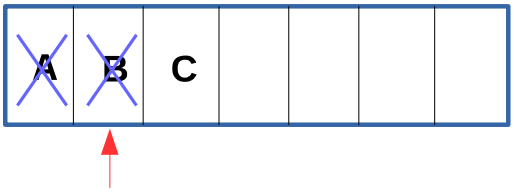
- Breadth first tree (BF Tree) after corresponding adjacent vertices have been discovered
- Graph after all vertices whose path length from the starting vertex is 1 have been found

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue





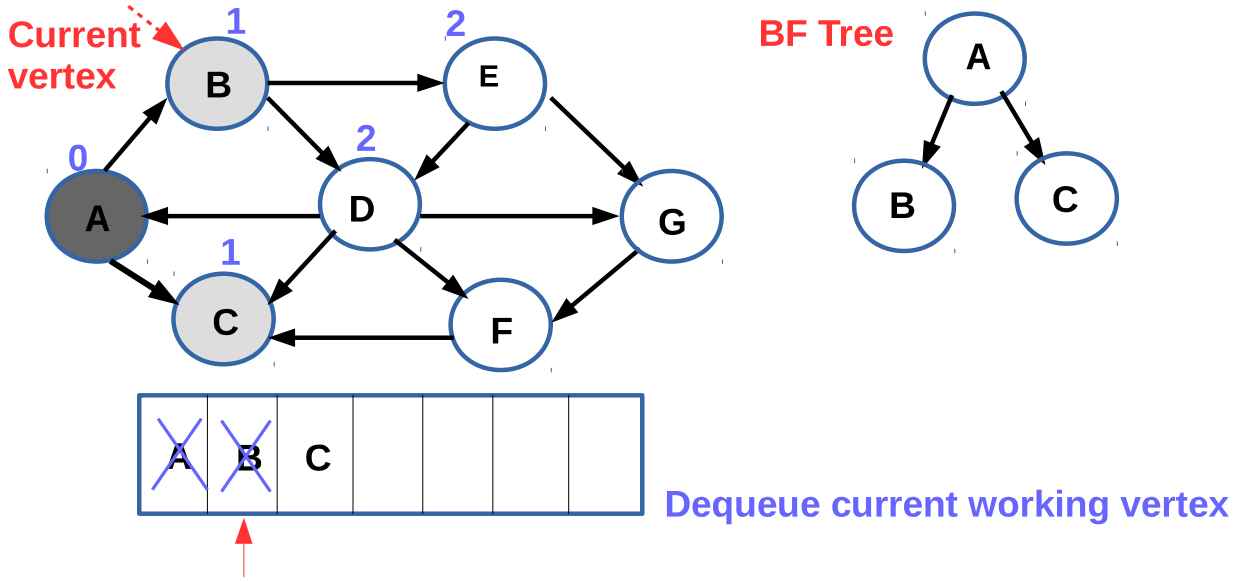


Dequeue current working vertex

- Darkest shaded vertices have already been completely processed
- Current “working” node (current vertex) is B

Breadth First Search - Example

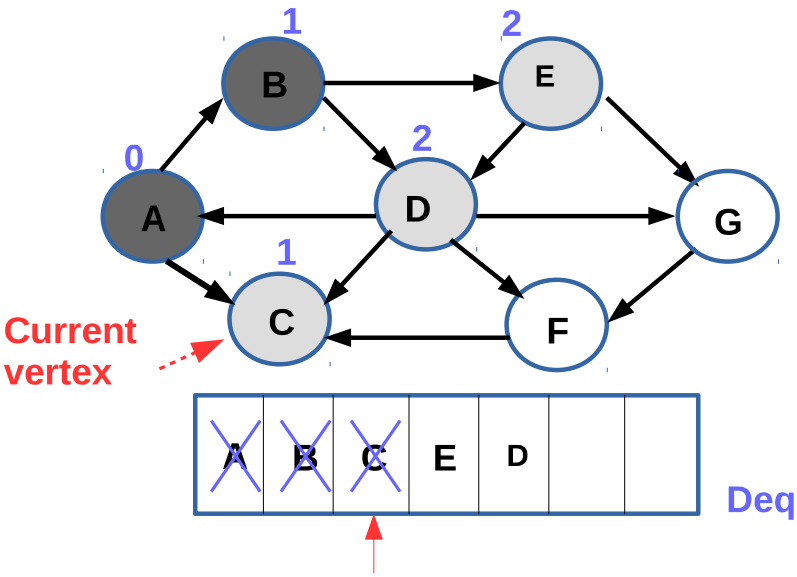
- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

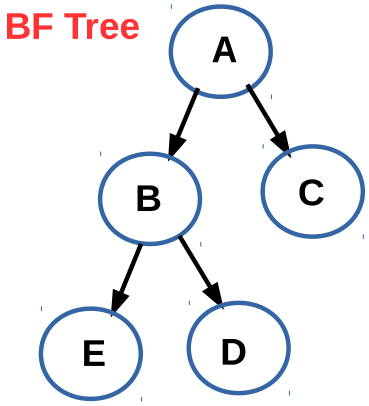


- Graph after all vertices whose path length from the starting vertex is 2 have been found

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

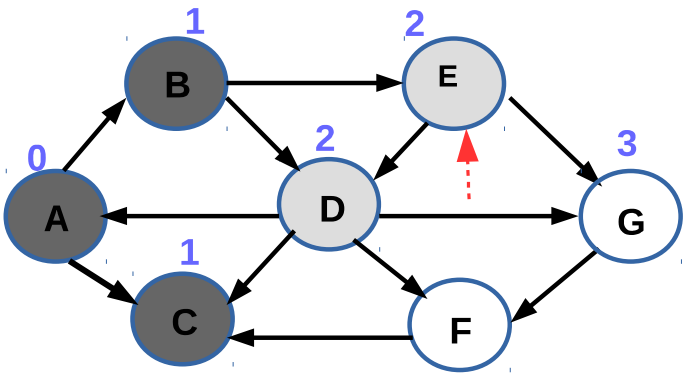


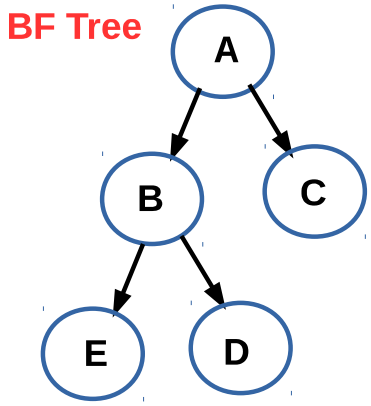


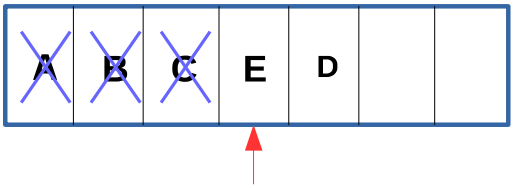
- Graph after all vertices whose path length from the starting vertex is 2 have been found
- Note: node C has no adjacent node(s). Continue the process

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.





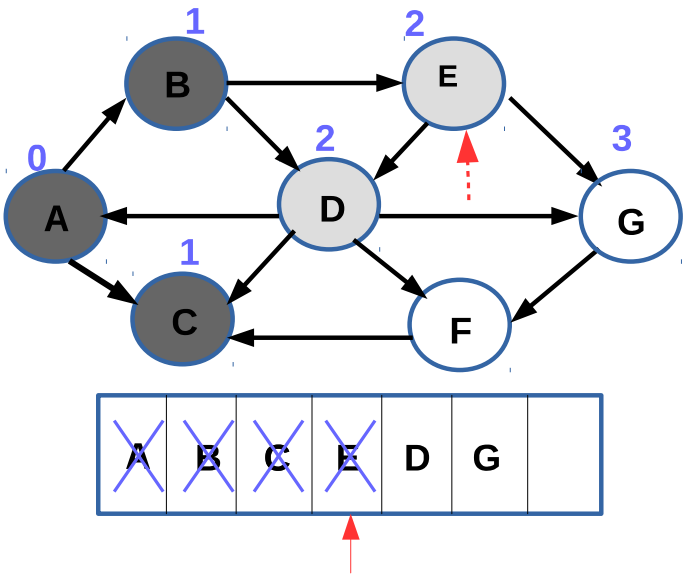


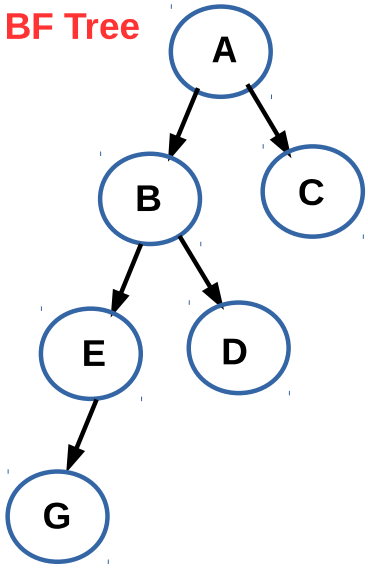
Dequeue current working vertex

- Note: node C has no adjacent node(s). Node C turns black. So continue the process
- Advance the pointer accordingly.

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

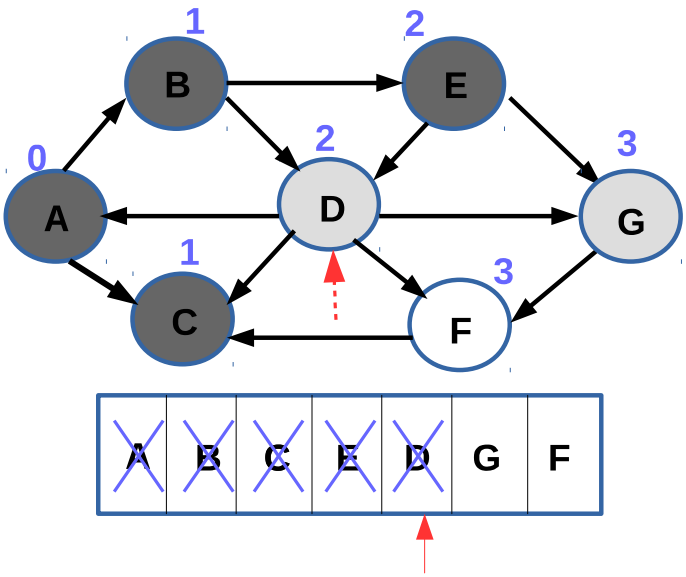


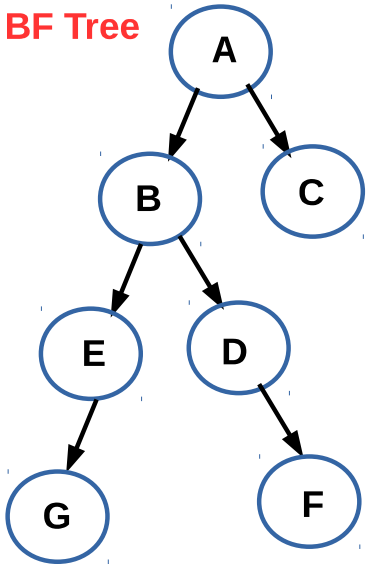


- Node E is the current working node, so <u>turns gray and dequeued</u>
- Enqueue (newly discovered) adjacent nodes to E and update tree

Breadth First Search - Example

- Graph after all vertices whose path length from the starting vertex is 3 have been found

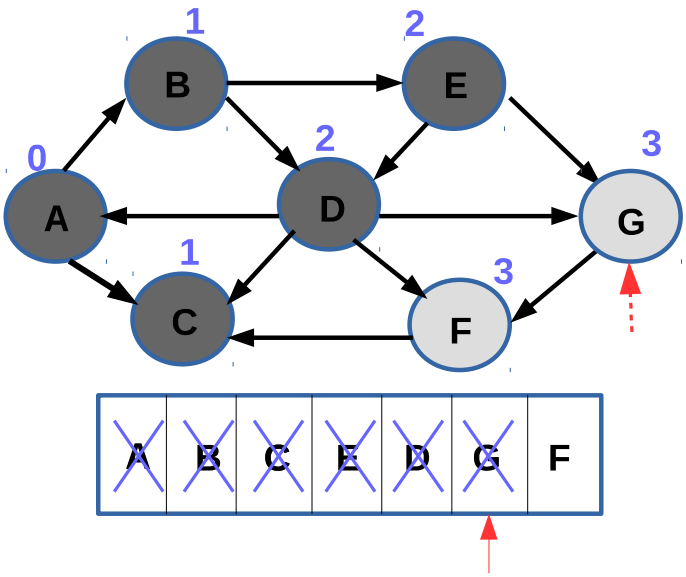


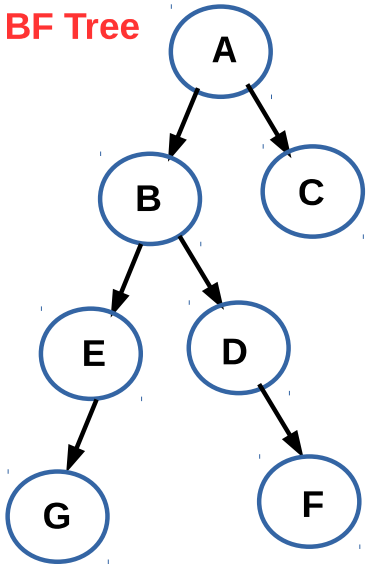


- Node D is the current working node, so <u>dequeued</u>
- Enqueue (newly discovered) adjacent nodes to D and update tree

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

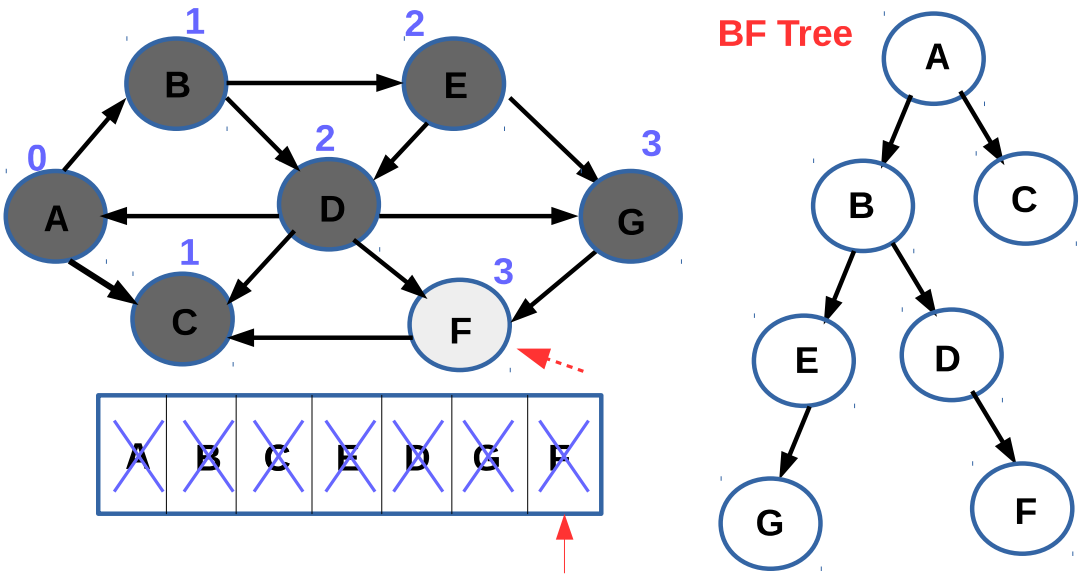




- Node G is the current working node, so turns gray and dequeued
- Enqueue (newly discovered) adjacent nodes to G (if any) and update tree

Breadth First Search - Example

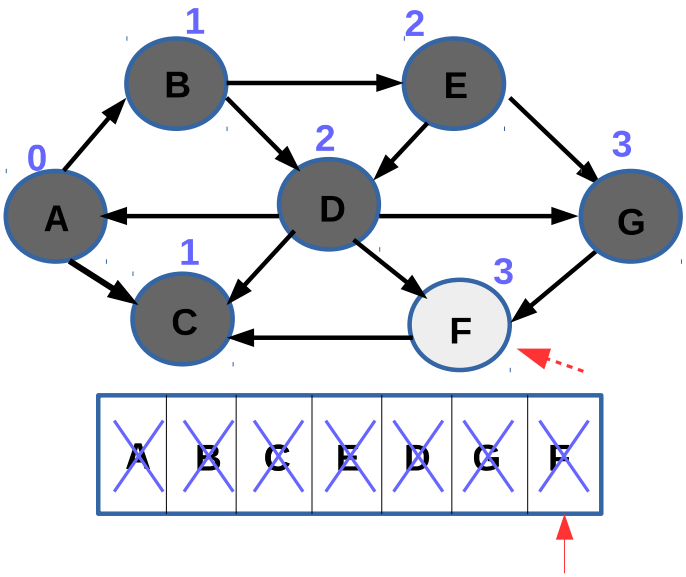
- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

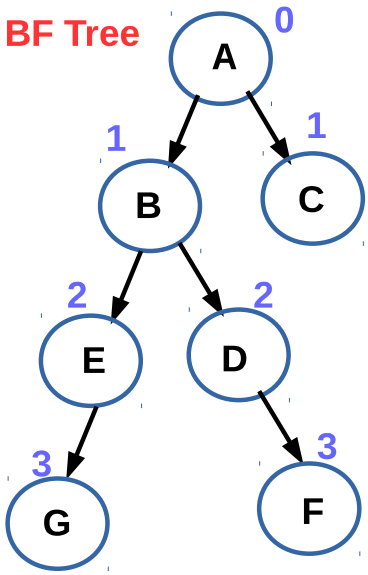


- By the time the pointer gets to F, we discover that **all nodes have been visited and the queue is empty**

Breadth First Search - Example

- Move pointer to the next item in the queue and check all adjacent vertices to the “current node” and enqueue accordingly.

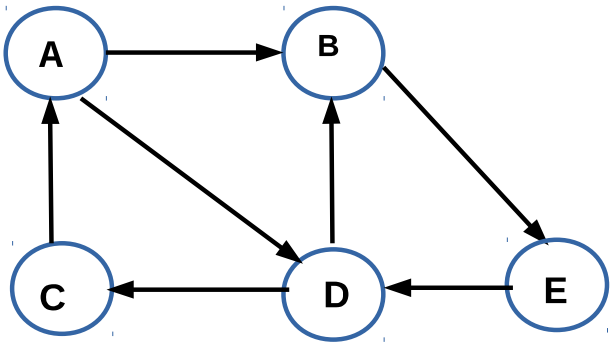




- By the time the pointer gets to F, we discover that **all nodes have been visited and the queue is empty**
- **Operation terminates once the queue is empty**

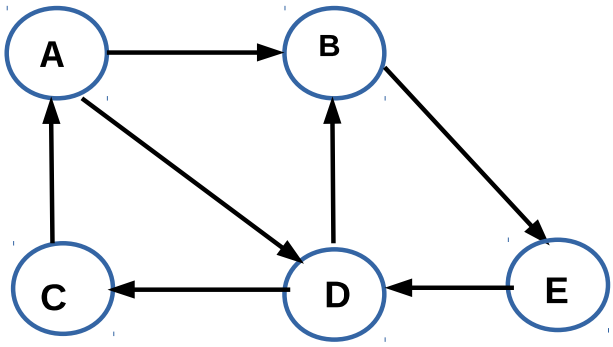
Breadth First Search – Exercise in Class

- Consider the directed graph below, use the breadth first search (BFS) procedure to give the BFS tree when A is the source node
- Show all intermediate steps with tree diagrams



Exercise in Class - Solution

- Consider the directed graph below, use the breadth first search (BFS) procedure to give the BFS tree when A is the source node
- Show each step of your operation

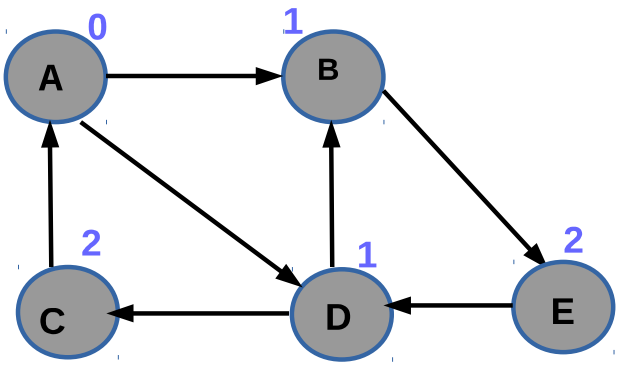


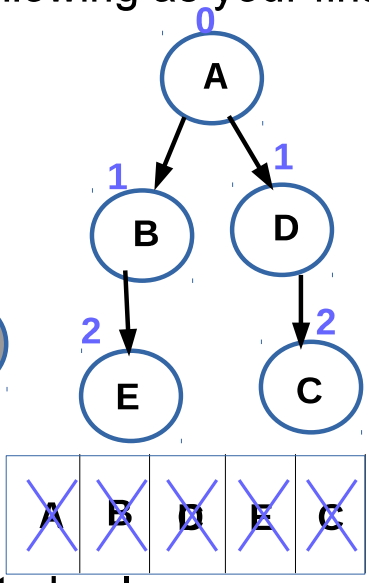
Source node

- **Note**: Follow the same procedure in the previous example

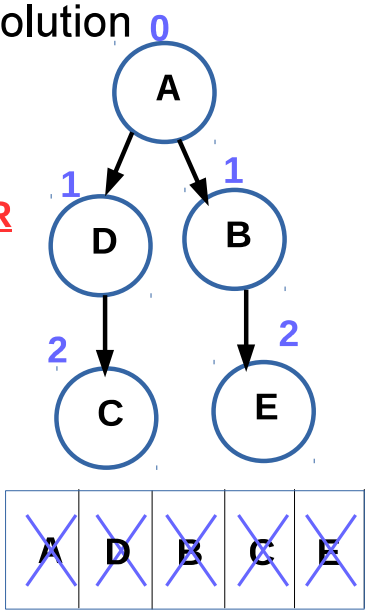
Exercise in Class - Solution

- If you follow the procedure correctly, when A is the source node
- You should have one of the following as your final solution





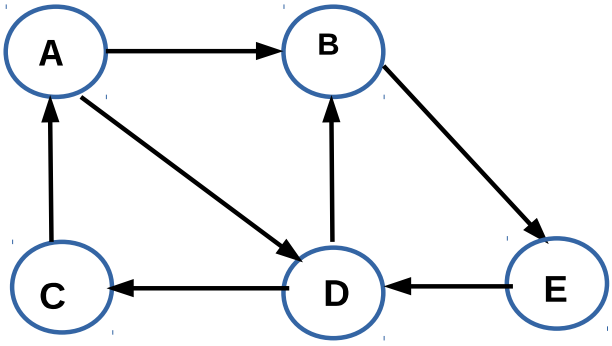
OR



- Note: all nodes have been visited only once
- Easy to compute shortest path with breadth first procedure (i.e. breadth first tree)

Exercise in Class -

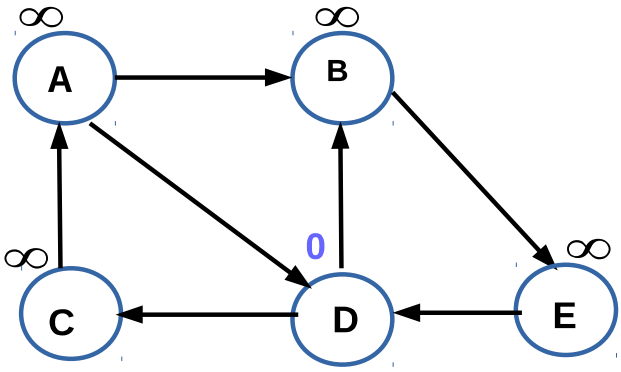
- Consider the directed graph below, use the breadth first search (BFS) procedure to give the BFS tree when **D** is the source node
- That is find the shortest unweighted path from D to all other nodes in the graph below



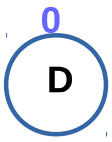
Source node

Exercise in Class - Solution

- Let the shortest path from D to any node be k and that k = ∞



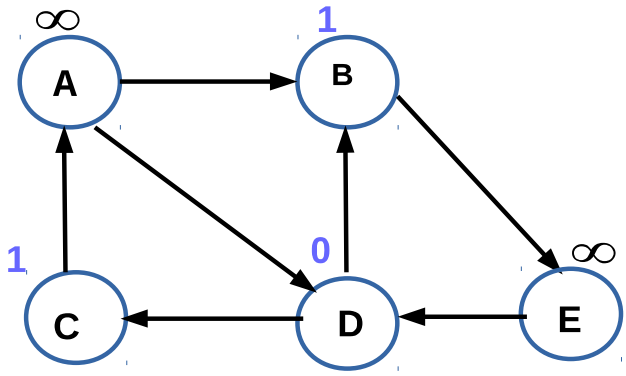
Directed Graph



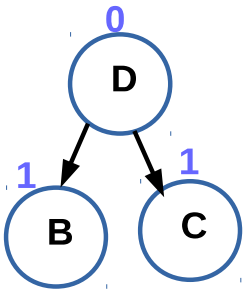
Breadth First Tree

Exercise in Class - Solution

- Let the shortest path from D to any node be k and that k = ∞
- Update each node's (unweighted) path length as at when discovered



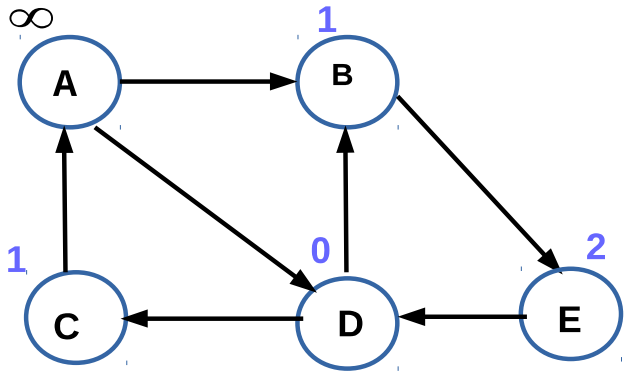
Directed Graph



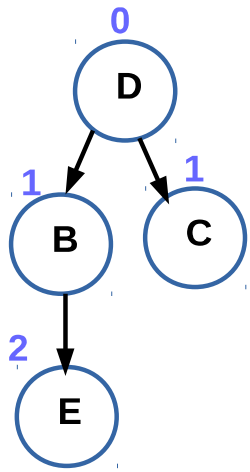
Breadth First Tree

Exercise in Class - Solution

- Let the shortest path from D to any node be k and that k = ∞
- Update each node's path length as at when discovered



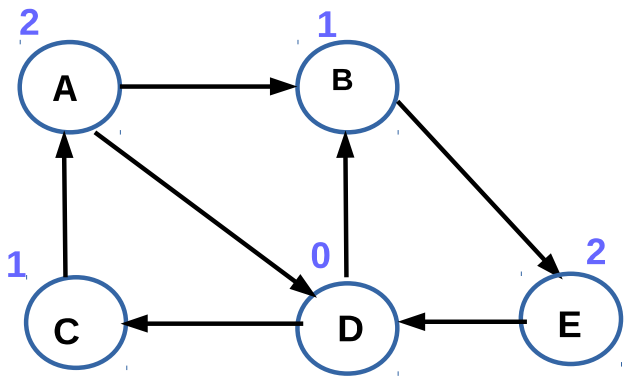
Directed Graph



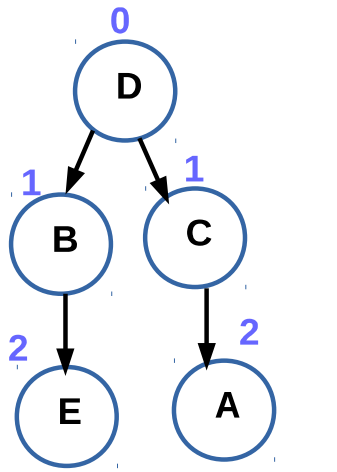
Breadth First Tree

Exercise in Class - Solution

- Let the shortest path from D to any node be k and that k = ∞
- Update each node's path length as at when discovered



Directed Graph



Breadth First Tree