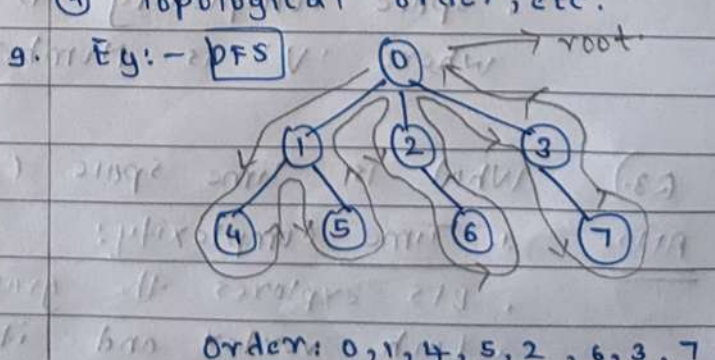
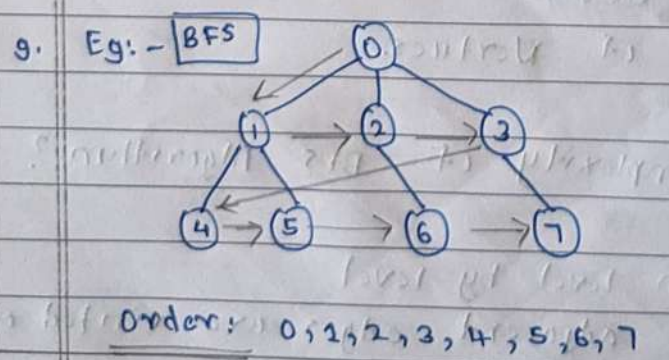


SPPU-TE-COMP-CONTENT – KSKA Git

QUESTION/ANSWERS

Q1.) Difference between DFS and BFS.
 ANS.: DFS : Depth First search.
 BFS : Breadth First search.

NO.	Breadth First Search (BFS)	NO.	Depth First Search (DFS)
1.	It is a vertex-based Algorithm	1.	It is a edge based Algorithm
2.	It uses '(Queue)' data structure.	2.	It uses '(stack)' data structure.
3.	It utilizes more memory and is inefficient.	3.	It utilizes less memory and is more efficient.
4.	Tree is wide and short.	4.	Tree is narrow and long.
5.	It traverses oldest unvisited vertices first.	5.	It traverses vertices along the edges in the beginning.
6.	Optimal for finding shortest distance, not in cost.	6.	Not optimal.
7.	Time complexity:- Adjacency matrix: $O(V^2)$ Adjacency lists: $O(V+E)$	7.	Time complexity:- Adjacency matrix: $O(V^2)$ Adjacency lists: $O(V+E)$
8.	Application: ① Examines Bipartite graph, ② Connected component, ③ Shortest path, etc	8.	Application: ① Examines 2 Edge connected graph, ② Strongly connected graph, ③ Acyclic graph, ④ Topological order, etc.



SPPU-TE-COMP-CONTENT - KSKA Git

Q2.) What is the Time and Space Complexity of DFS Algorithm?

ANS. ① DFS or Depth First Search is a traversal approach for a graph in which we traverse all adjacent vertices one by one.

② The Algorithm starts from a given source and explores all reachable vertices from the given source.

Time Complexity:

i) Using Adjacency List:

The time complexity is given by $O(V + E)$ where

V is the Number of vertices

E is the Number of Edges.

This is only because in DFS every vertex is visited once and each edge is explored once.

ii) Using Adjacency Matrix:

Time Complexity will be $O(V^2)$ where V is the Number of Vertices.

Space Complexity:

i) Using Adjacency List:

The space complexity of DFS is $O(V)$ for stack used for iterative (recursive) implementation.

The space required for storing the graph is $O(V + E)$ where V is the Number of vertices, and E number of Edges in the Adjacency List.

ii) Using Adjacency Matrix:

space complexity is $O(V^2)$

where V is Number of Vertices.

Q3.) What is Time Space Complexity of BFS Algorithm?

ANS. i. Time Complexity:

BFS explores the graph level by level

Each vertex and its adjacent edges are visited only once.

SPPU-TE-COMP-CONTENT – KSKA Git

• Therefore, In worst case every edge will be explored and the Time complexity will be $O(V+E)$

2. Space Complexity:

- The primary space usage comes from the Queue
- In the worst case, all vertices at a particular level might be enqueued simultaneously
- Hence, The space complexity of Breadth First Search (BFS) is $O(V)$

Conclusion :-

Hence, We implemented the two traversal algorithms i.e. Depth First Search (DFS) and Breadth First Search (BFS) using a recursive approach (iterative) to traverse a graph.