Batch: D3	**Roll No.: 16010123294**
Experiment / assignment / tutorial No. 08	
Grade: AA / AB / BB / BC / CC / CD /DD	
Signature of the Staff In-charge with date	

Title: Study of Graph traversal methods

Objective: To Understand Graph Traversal Methods - BFS & DFS.

Expected Outcome of Experiment:

CO	Outcome
2	Apply linear and non-linear data structure in application development.

Books/ Journals/ Websites referred:

1. *Fundamentals Of Data Structures In C* – Ellis Horowitz, Satraj Sahni, Susan Anderson-Fred
2. *An Introduction to data structures with applications* – Jean Paul Tremblay, Paul G. Sorenson
3. *Data Structures A Pseudo Approach with C* – Richard F. Gilberg & Behrouz A. Forouzan
4. https://www.geeksforgeeks.org/binary-tree-data-structure/
5. https://www.thecrazyprogrammer.com/2015/03/c-program-for-binary-search-tree-insertion.html

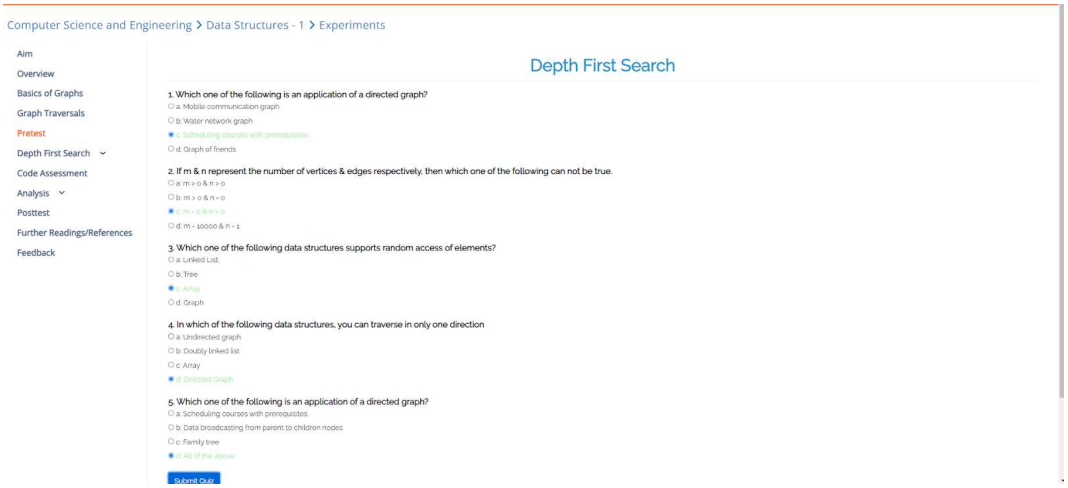
Abstract:

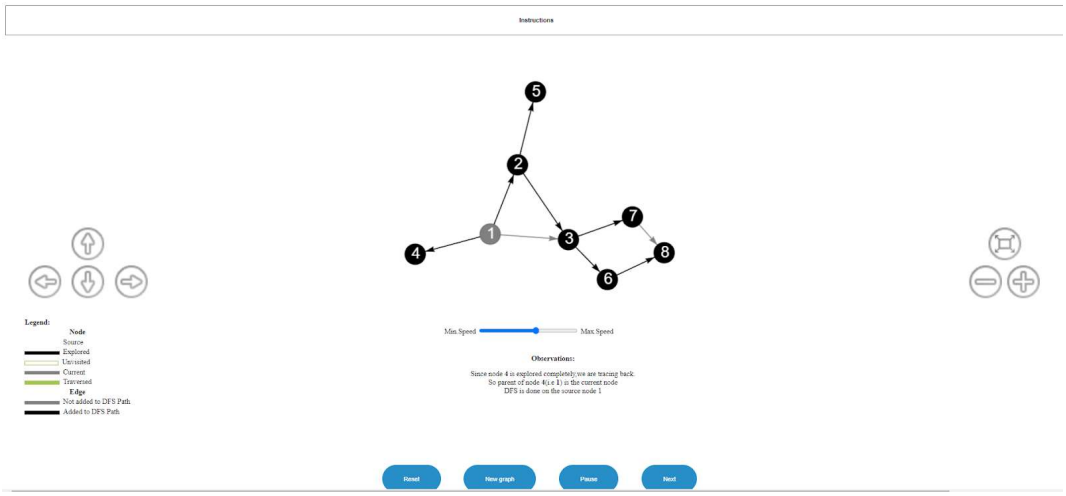
Graph is a non-linear data structure consisting of vertices and edges. The vertices are sometimes also referred to as nodes and the edges are lines or arcs that connect any two nodes in the graph. More formally a Graph is composed of a set of vertices(V) and a set of edges(E). The graph is denoted by G(V, E).

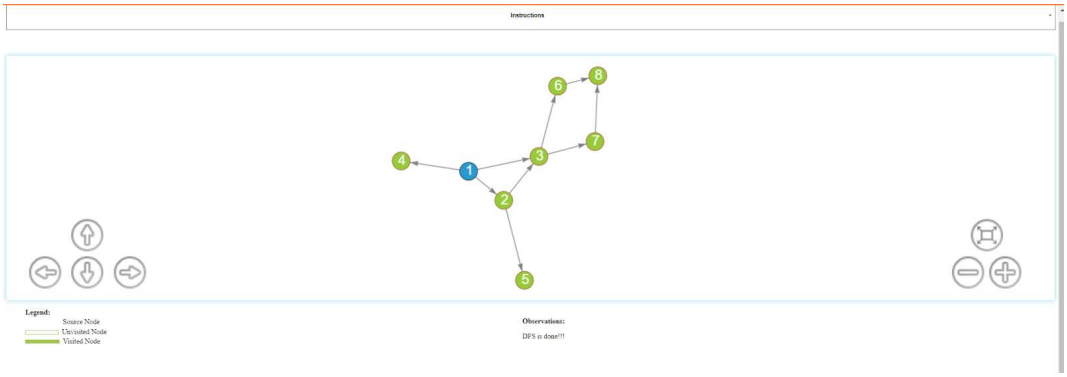
Graph data structures are a powerful tool for representing and analyzing complex relationships between objects or entities. They are particularly useful in fields such as social network analysis, recommendation systems, and computer networks. In the field of sports data science, graph data structures can be used to analyze and understand the dynamics of team performance and player interactions on the field.

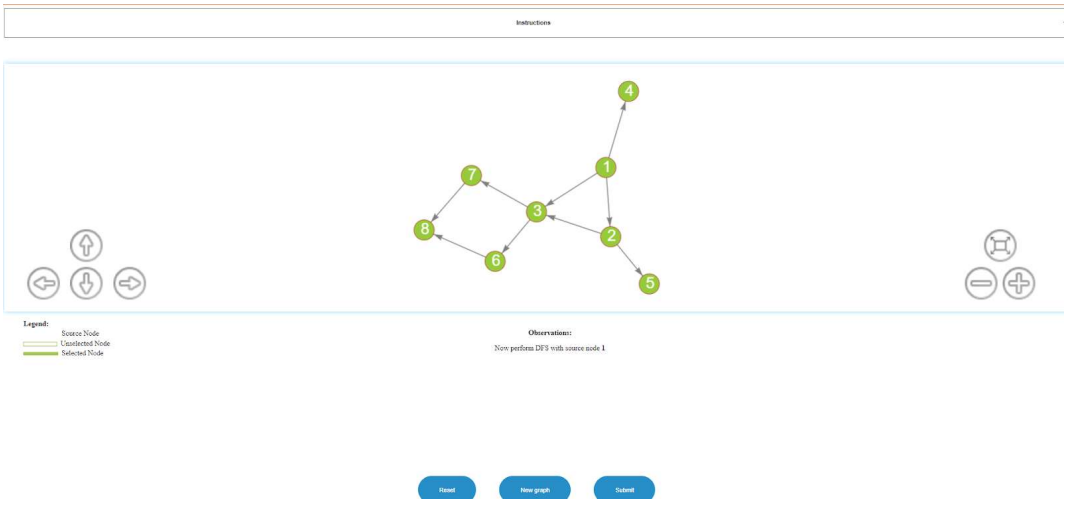
BFS & DFS on the assigned graphs:

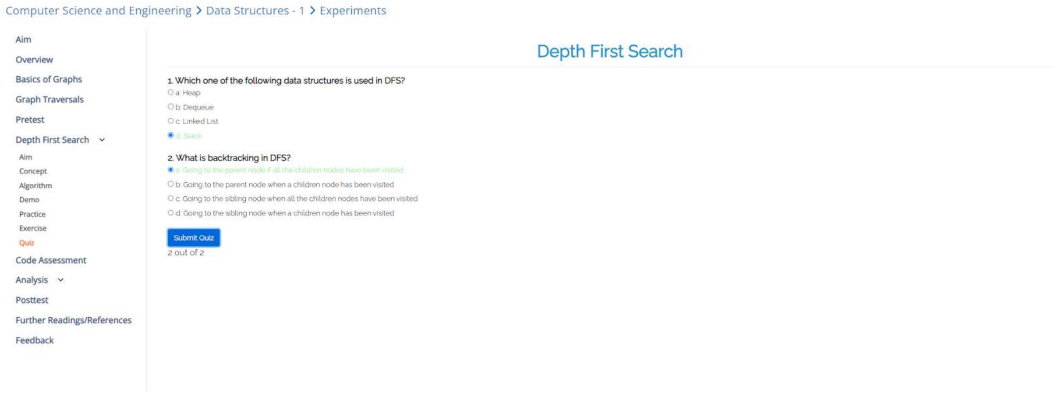
DFS

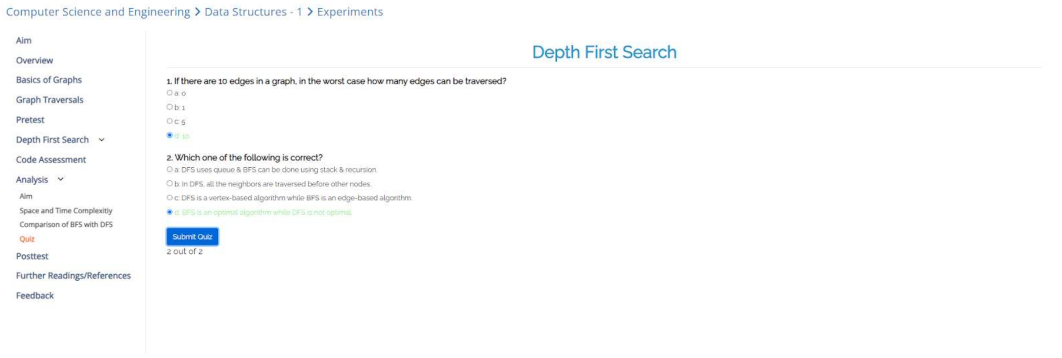


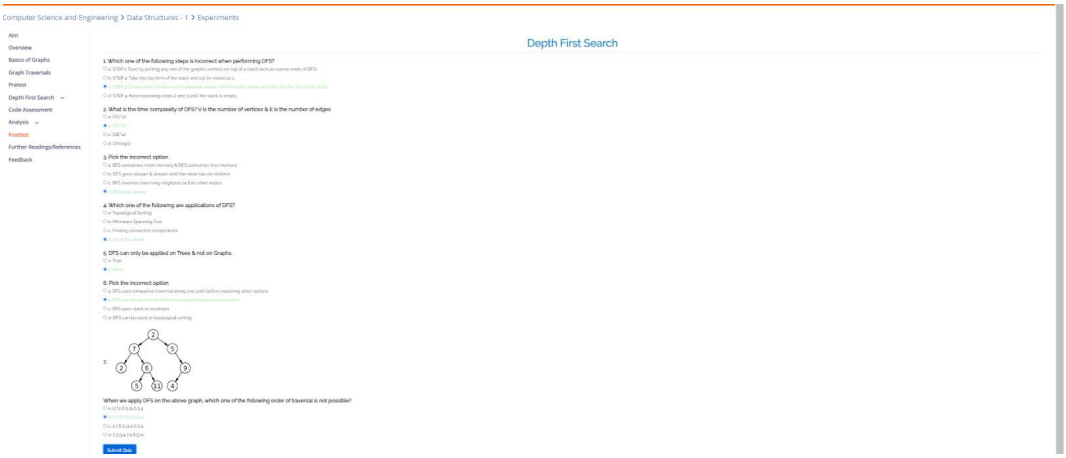

Instructions
Legend:
Node
Source
Explored
Unvisited
Current
Traversed
Edge
Not added to DFS Path
Added to DFS Path
Min Speed
Max Speed
Observations:
Since node 4 is explored completely,we are tracing back
So parent of node 4(i.e 1) in the current node
DFS is done on the source node 1
Reset
New graph
Pause
Next


Instructions
Legend:
Source Node
Unvisited Node
Visited Node
Observations:
DFS is done!!!

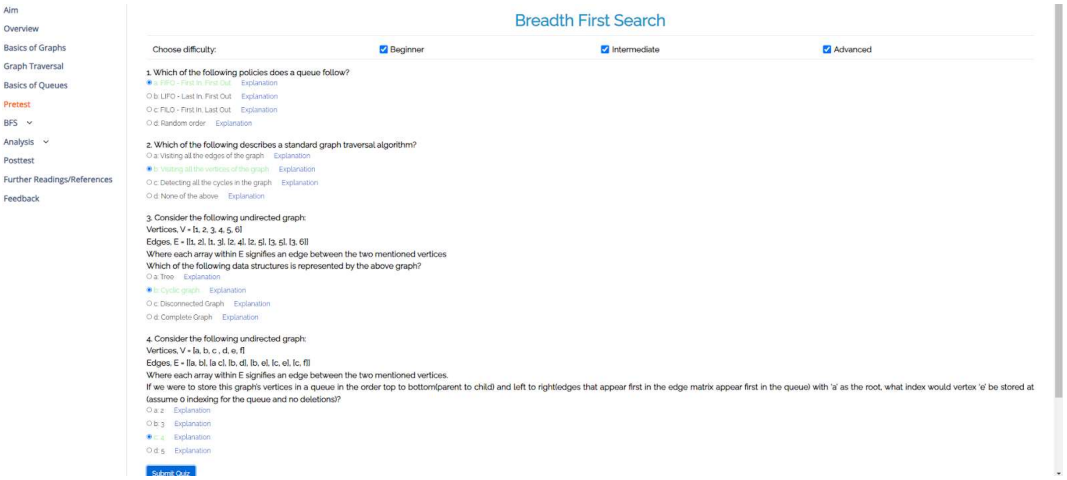

Instructions
Legend:
Source Node
Unselected Node
Selected Node
Observations:
Now perform DFS with source node 1.
Reset
New graph
Submit

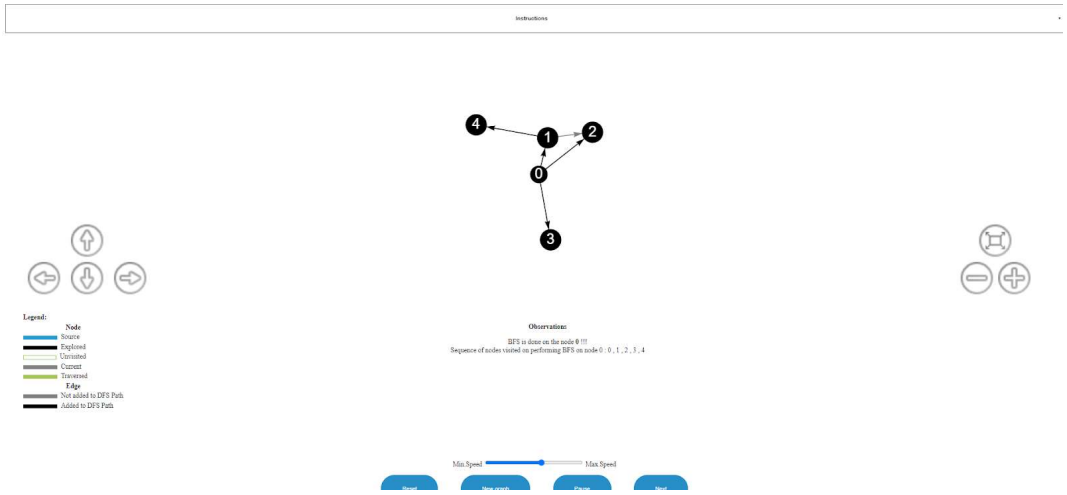

Computer Science and Engineering > Data Structures - 1 > Experiments
Aim
Overview
Basics of Graphs
Graph Traversals
Pretest
Depth First Search
Aim
Concept
Algorithm
Demo
Practice
Exercise
Quiz
Code Assessment
Analysis
Posttest
Further Readings/References
Feedback
Depth First Search
1. Which one of the following data structures is used in DFS?
a. Heap
b. Dequeue
c. Linked List
d. Stack
2. What is backtracking in DFS?
a. Going to the parent node if all the children nodes have been visited
b. Going to the parent node when a children node has been visited
c. Going to the sibling node when all the children nodes have been visited
d. Going to the sibling node when a children node has been visited
Submit Quiz
2 out of 2

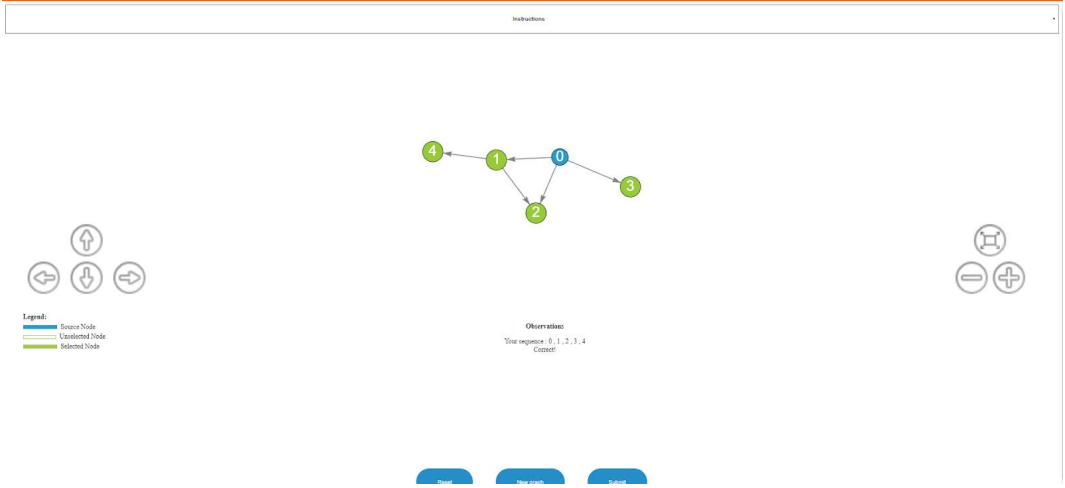

Computer Science and Engineering > Data Structures - 1 > Experiments
Aim
Overview
Basics of Graphs
Graph Traversals
Pretest
Depth First Search
Code Assessment
Analysis
Aim
Space and Time Complexity
Comparison of BFS with DFS
Quiz
Posttest
Further Readings/References
Feedback
Depth First Search
1. If there are 10 edges in a graph, in the worst case how many edges can be traversed?
a. 0
b. 1
c. 5
d. 10
2. Which one of the following is correct?
a. DFS uses queue & BFS can be done using stack & recursion
b. In DFS, all the neighbors are traversed before other nodes
c. DFS is a vertex-based algorithm while BFS is an edge-based algorithm
d. BFS is an optimal algorithm while DFS is not optimal
Submit Quiz
2 out of 2

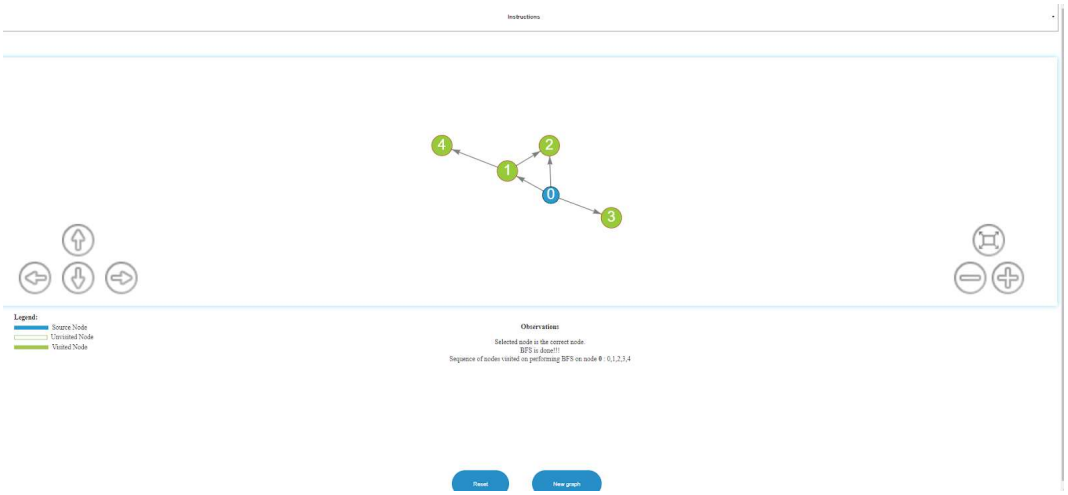

Computer Science and Engineering > Data Structures - 1 > Experiments
Depth First Search

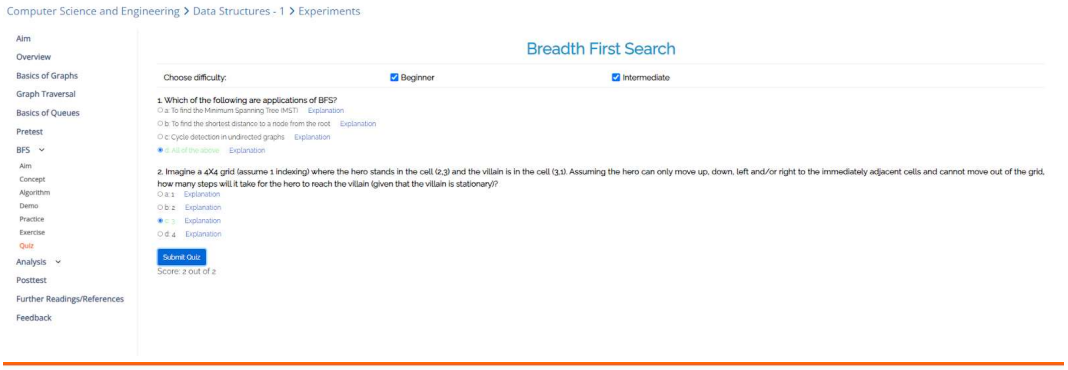
BFS

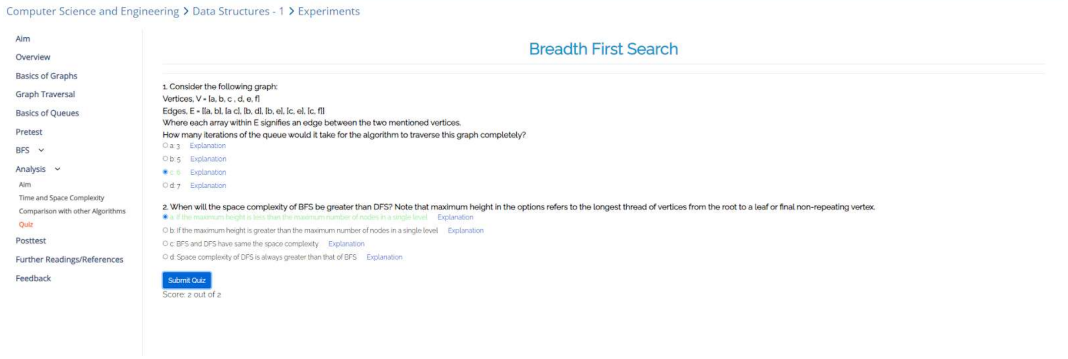

Aim
Overview
Basics of Graphs
Graph Traversal
Basics of Queues
Pretest
BFS
Analysis
Posttest
Further Readings/References
Feedback
Breadth First Search
Choose difficulty:
Beginner
Intermediate
Advanced
1. Which of the following policies does a queue follow?
b. LIFO - Last In, First Out
c. FILO - First In, Last Out
d. Random order
2. Which of the following describes a standard graph traversal algorithm?
a. Visiting all the edges of the graph
b. Visiting all the vertices of the graph
c. Detecting all the cycles in the graph
d. None of the above
3. Consider the following undirected graph:
Vertices, V = {1, 2, 3, 4, 5, 6}
Edges, E = {{1, 2}, {1, 3}, {2, 4}, {2, 5}, {3, 5}, {3, 6}}
Where each array within E signifies an edge between the two mentioned vertices
Which of the following data structures is represented by the above graph?
a. Tree
b. Cyclic graph
c. Disconnected Graph
d. Complete Graph
4. Consider the following undirected graph:
Vertices, V = {a, b, c , d, e, f}
Edges, E = {{a, b}, {a c}, {b, d}, {b, e}, {c, e}, {c, f}}
Where each array within E signifies an edge between the two mentioned vertices.
If we were to store this graph's vertices in a queue in the order top to bottom(parent to child) and left to right(edges that appear first in the edge matrix appear first in the queue) with 'a' as the root, what index would vertex 'e' be stored at (assume 0 indexing for the queue and no deletions)?
a. 2
b. 3
c. 4
d. 5
Submit Quiz

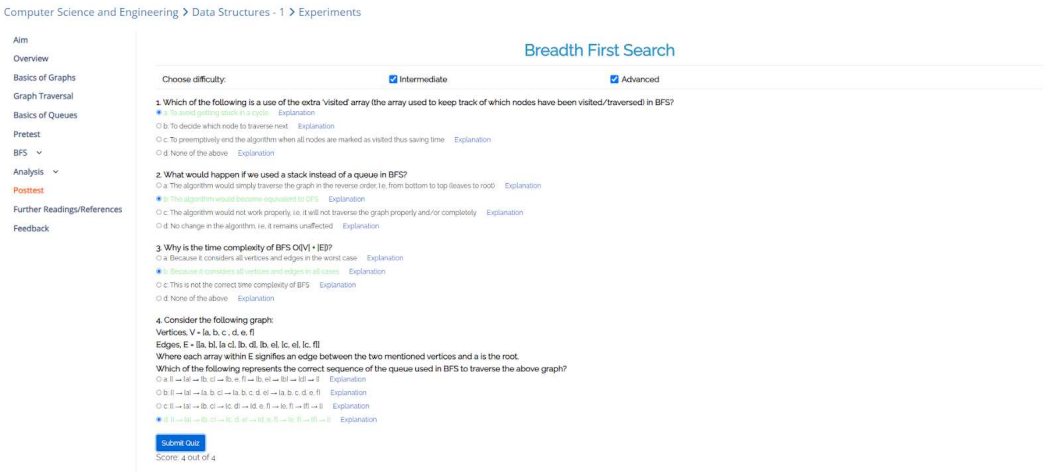

Instructions
Legend:
Node
Source
Explored
Unvisited
Current
Traversed
Edge
Not added to DFS Path
Added to DFS Path
Observation:
BFS is done on the node 0 !!!
Sequence of nodes visited on performing BFS on node 0 : 0 , 1 , 2 , 3 , 4
Min Speed
Max Speed


Instructions
Legend:
Source Node
Unselected Node
Selected Node
Observation:
Your sequence : 0 , 1 , 2 , 3 , 4
Correct!


Instructions
4
2
1
0
3
Legend:
Source Node
Unvisited Node
Visited Node
Observation:
Selected node is the correct node.
BFS is done!!!
Sequence of nodes visited on performing BFS on node 0 : 0,1,2,3,4
Reset
New graph


Computer Science and Engineering > Data Structures - 1 > Experiments
Aim
Overview
Basics of Graphs
Graph Traversal
Basics of Queues
Pretest
BFS
Aim
Concept
Algorithm
Demo
Practice
Exercise
Quiz
Analysis
Posttest
Further Readings/References
Feedback
Breadth First Search
Choose difficulty:
Beginner
Intermediate
1. Which of the following are applications of BFS?
a. To find the Minimum Spanning Tree (MST) Explanation
b. To find the shortest distance to a node from the root Explanation
c. Cycle detection in undirected graphs Explanation
d. All of the above Explanation
2. Imagine a 4X4 grid (assume 1 indexing) where the hero stands in the cell (2,3) and the villain is in the cell (3,1). Assuming the hero can only move up, down, left and/or right to the immediately adjacent cells and cannot move out of the grid, how many steps will it take for the hero to reach the villain (given that the villain is stationary)?
a. 1 Explanation
b. 2 Explanation
c. 3 Explanation
d. 4 Explanation
Submit Quiz
Score: 2 out of 2


Computer Science and Engineering > Data Structures - 1 > Experiments
Aim
Overview
Basics of Graphs
Graph Traversal
Basics of Queues
Pretest
BFS
Analysis
Aim
Time and Space Complexity
Comparison with other Algorithms
Quiz
Posttest
Further Readings/References
Feedback
Breadth First Search
1. Consider the following graph:
Vertices, V = {a, b, c , d, e, f}
Edges, E = {{a, b}, {a c}, {b, d}, {b, e}, {c, e}, {c, f}}
Where each array within E signifies an edge between the two mentioned vertices.
How many iterations of the queue would it take for the algorithm to traverse this graph completely?
a. 3 Explanation
b. 5 Explanation
c. 6 Explanation
d. 7 Explanation
2. When will the space complexity of BFS be greater than DFS? Note that maximum height in the options refers to the longest thread of vertices from the root to a leaf or final non-repeating vertex.
a. If the maximum height is less than the maximum number of nodes in a single level Explanation
b. If the maximum height is greater than the maximum number of nodes in a single level Explanation
c. BFS and DFS have same the space complexity Explanation
d. Space complexity of DFS is always greater than that of BFS Explanation
Submit Quiz
Score: 2 out of 2



Conclusion:-We implement the BFS and DFS via virtual lab for deeper understanding.

PostLab Questions:

1) Differentiate between BFS & DFS

Traversal Approach:

BFS: Explores neighbors level by level, moving outward from the starting node.

DFS: Explores as far down a branch as possible before backtracking.

Data Structure:

BFS: Uses a queue to manage nodes, ensuring a level-order traversal.

DFS: Uses a stack (or recursion) to go deep into each branch before returning.

Time Complexity:

Both BFS and DFS have a time complexity of O(V + E), where VVV is the number of vertices and EEE is the number of edges.

Applications:

BFS: Ideal for finding the shortest path in unweighted graphs.

DFS: Useful for cycle detection, pathfinding in mazes, and topological sorting.

2) Applications of BFS & DFS

BFS Applications:

- Shortest Path in Unweighted Graphs: Finds the shortest path between two nodes in an unweighted graph.
- Web Crawlers: Crawls web pages level by level.
- Social Networking Sites: Finds connections or friends within a certain degree.
- GPS Navigation: Helps in finding the shortest route in maps (unweighted scenarios).

DFS Applications:

- Cycle Detection: Identifies cycles in directed and undirected graphs.
- Topological Sorting: Used in scheduling tasks where certain tasks depend on others.
- Maze/Pathfinding Algorithms: Helps in exploring all possible paths and backtracking if a path is blocked.
- Solving Puzzles: Solves problems like Sudoku, where depth exploration is needed to find a solution.