

Assignment 1

DFS and BFS Algorithms

⇒ Problem Statement: Develop and implement recursive algorithms for searching all the vertices of a tree using depth first search (DFS) and breadth first search (BFS).

⇒ Theory:

If one wants to go from point A to point B, you are employing some kind of search. For a chess game, point A to point B might be two points between its current position & its position 5 moves from now. Searching falls under Artificial Intelligence (AI).

1) Depth First Search (DFS): DFS is an algorithm for traversing a tree or graph data structure. The algorithm starts at the root node and explores as far as possible along each branch before backtracking. So, the basic idea is to start from the root and mark the node and move to adjacent unmarked node and continue until there is no unmarked adjacent node.

* Pseudo code for DFS:

1) Declare two empty lists: open and closed.

2) Add start node to our open list

3) While our open list is not empty, loop the following:

a. Remove the first node from open list

b. Check if the removed node is our target node

- If it is target, break the loop and return success

- Else, go to step c

c. Extract the neighbors of above removed node

d. Add the neighbors to our open list, and removed node to closed.

e. Continue looping

2) Breadth first Search (BFS): BFS is a traversing algorithm where traversal starts from a selected node and goes layer-wise thus exploring the neighbour nodes (nodes which are

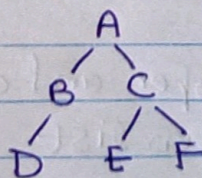
directly connected to source node. You must then move towards the next level neighbour nodes.

* Pseudocode for BFS:

- 1) Declare two empty lists: open and closed
- 2) Add start node to our open list
- 3) While open list is not empty, do the following steps:
 - a. Remove the first node from open list
 - b. Check if the removed node is our target node:
 - If target, then return success
 - Else, continue step c.
 - c. Extract the neighbours of removed node
 - d. Add the neighbours to end of our open list and the removed node to our closed list.

* Example:

Let the tree be:



DFS Traversal: A, B, D, C, E, F

BFS Traversal: A, B, C, D, E, F

⇒ Conclusion: Recursive algorithms for searching all the vertices of a tree using Depth First Traversal (DFS) and Breadth First Search (BFS) were implemented.