

Lab Activity - 1

Steps of DFS Algorithm used here:

1. Initialize a stack and push the starting position along with the path containing only the start.
2. Create a set to stored visited nodes.
3. While the stack is not empty:
 - a. Pop the top element (LIFO).
 - b. If it is the goal, return the path.
 - c. If not visited:
 - i. Mark as visited
 - ii. Explore all valid neighbour cells (up, down, left, right)
 - iii. Push each valid neighbour onto the stack with the updated path.
4. If the stack becomes empty and goal not found, return None (no path exists).

Why DFS works here :-

- DFS explores along the path.
- Uses stack.
- Does not visit cycles.
- Uses less memory.

- DFS explores as far as possible along one branch before backtracking
- Uses stack (LIFO).
- Does not guarantee shortest path.
- Uses backtracking naturally via stack pop().