Name: Saira Irshad

Roll no: SU92-BSAIM-f24\_121

Submitted To: Sir Rasik Ali

Section: BSAI-3b

**Depth First Search (DFS) using Stack in Artificial Intelligence**

Depth First Search (DFS) is a fundamental graph traversal algorithm used in Artificial Intelligence. It explores as far as possible along a branch before backtracking. The stack data structure is used to manage the nodes to be explored. In this document, we explain DFS using an iterative stack-based approach.

# 1. How DFS with Stack Works

In this implementation, we use a stack (LIFO - Last In, First Out) to control the order of traversal:  
1. Push the start node into the stack.  
2. Pop the top element from the stack.  
3. If it is not visited, mark it as visited and process it.  
4. If it matches the goal node, stop the search.  
5. Otherwise, push all its unvisited neighbors into the stack.  
6. Repeat until the stack is empty.

# 2. DFS with Stack - Code Example

Here is the Python code implementation:

def dfs\_stack(graph, start, goal):  
 visited = set()   
 stack = [start]   
  
 while stack:  
 element = stack.pop()   
 if element not in visited:  
 print(element)   
 visited.add(element)  
  
 if element == goal:  
 print(f"Goal {goal} found!")  
 return visited  
  
 stack.extend(graph[element] - visited)  
  
 return visited  
  
graph = {  
 '0': set(['1', '2']),  
 '1': set(['0', '3', '4']),  
 '2': set(['0']),  
 '3': set(['1']),  
 '4': set(['2', '3'])  
}  
  
dfs\_stack(graph, '0', '3')

# 3. Advantages and Disadvantages

* ✔ Advantages:
* • Simple to implement with stack.
* • Good for searching deep paths.
* • Requires less memory compared to BFS in many cases.
* ❌ Disadvantages:
* • May get stuck in deep or infinite paths if not handled properly.
* • Does not guarantee the shortest path.

# Conclusion

DFS using stack is an important algorithm for exploring graphs in Artificial Intelligence. It explores depth before breadth, making it suitable for problems where deeper solutions are likely. However, it does not always guarantee the shortest path like BFS.