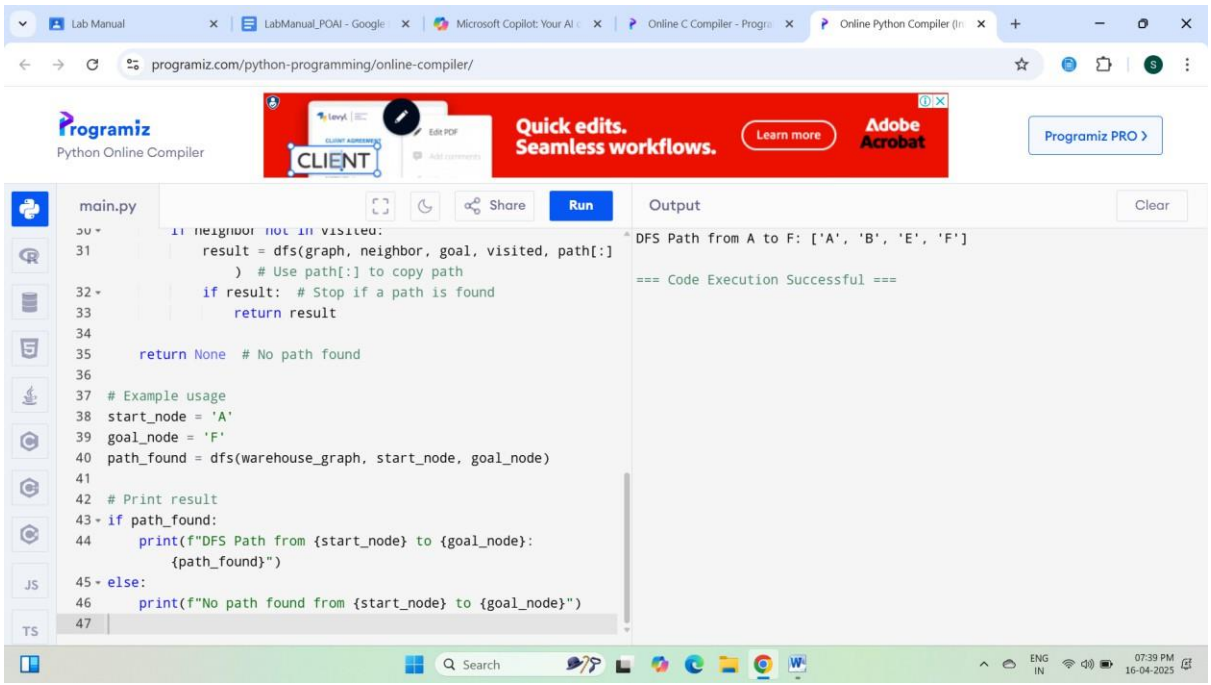
PRINCIPLES OF ARTIFICIAL INTELLIGENCE
ASSIGNMENT -2
NAME: SUJIT R.
REG NO: 241801280

```
# Depth First Search (DFS) implementation for a warehouse graph
# Sample warehouse graph as an adjacency list

warehouse_graph = {

    'A': ['B', 'C'],

    'B': ['D', 'E'],

    'C': ['F'],

    'D': [],

    'E': ['F'],

    'F': []

}


# Function to perform DFS

def dfs(graph, start, goal, visited=None, path=None):

    if visited is None:

        visited = set()

    if path is None:

        path = []


    # Mark current node as visited and add to path

    visited.add(start)

    path.append(start)


    # If goal is found, return the path

    if start == goal:

        return path
```

```
    # Explore neighbors
    for neighbor in graph.get(start, []):  # Ensure 'start' exists in graph
        if neighbor not in visited:
            result = dfs(graph, neighbor, goal, visited, path[:])  # Use path[:] to copy path
            if result:  # Stop if a path is found
                return result

    return None  # No path found

# Example usage
start_node = 'A'
goal_node = 'F'
path_found = dfs(warehouse_graph, start_node, goal_node)

# Print result
if path_found:
    print(f"DFS Path from {start_node} to {goal_node}: {path_found}")
else:
    print(f"No path found from {start_node} to {goal_node}")
```


Lab Manual
LabManual_POAI - Google
Microsoft Copilot: Your AI
Online C Compiler - Progra
Online Python Compiler (In
programiz.com/python-programming/online-compiler/
Programiz
Python Online Compiler
Quick edits.
Seamless workflows.
Learn more
Adobe Acrobat
Programiz PRO
main.py
Share
Run
Output
Clear
result = dfs(graph, neighbor, goal, visited, path[:]
) # Use path[:] to copy path
if result: # Stop if a path is found
return result
return None # No path found
# Example usage
start_node = 'A'
goal_node = 'F'
path_found = dfs(warehouse_graph, start_node, goal_node)
# Print result
if path_found:
print(f"DFS Path from {start_node} to {goal_node}:
{path_found}")
else:
print(f"No path found from {start_node} to {goal_node}")
DFS Path from A to F: ['A', 'B', 'E', 'F']
=== Code Execution Successful ===
Search
ENG
IN
07:39 PM
16-04-2025