Experiment 9: DEADTH FIRST SEARCH

AIM: Implement an Algorithm in Python to solve the DFS problem.

CODE:

```
def dfs(graph, start, goal, visited=None, path=None):
    if visited is None:
        visited = set()
    if path is None:
        path = []

    visited.add(start)
    path = path + [start]

    if start == goal:
        return path

    for neighbor in graph[start]:
        if neighbor not in visited:
            new_path = dfs(graph, neighbor, goal, visited, path)
            if new_path:
                return new_path

    return None

# Example usage
graph = {
    'A': ['B', 'C'],
```

```
    'B': ['A', 'D', 'E'],
    'C': ['A', 'F'],
    'D': ['B'],
    'E': ['B', 'F'],
    'F': ['C', 'E']
}

start_node = 'A'
goal_node = 'F'

path = dfs(graph, start_node, goal_node)
print("Path:", path)
```

OUTPUT:

Path: ['A', 'B', 'D', 'E', 'F']

RESULT: Code has been Implemented successfully.