1.B.)Solve any problem using depth first search

AIM:

To determine whether a path exists from a starting node to a target node in a graph using Depth First Search (DFS) traversal.

CODE:

```
def dfs(graph, start, visited=None):
    if visited is None:
        visited = set()

    visited.add(start)
    result = [start]

    for neighbor in graph.get(start, []):
        if neighbor not in visited:
            result.extend(dfs(graph, neighbor, visited))

    return result
graph = {
    'A': ['B', 'C'],
    'B': ['D', 'E'],
    'C': ['F'],
    'D': [],
    'E': ['F'],
    'F': []
}

start_node = 'A'
visited_nodes = dfs(graph, start_node)

print(f"DFS traversal starting from {start_node}: {visited_nodes}")
```

OUTPUT:

```
    DFS traversal starting from A: ['A', 'B', 'D', 'E', 'F',
'C']
```

RESULT:

The code is executed as expected and the output have been verified successfully.