



National University of Modern Languages

Artificial Intelligence - LAB

Lab # 7

BSSE 5(M)

Submitted By:

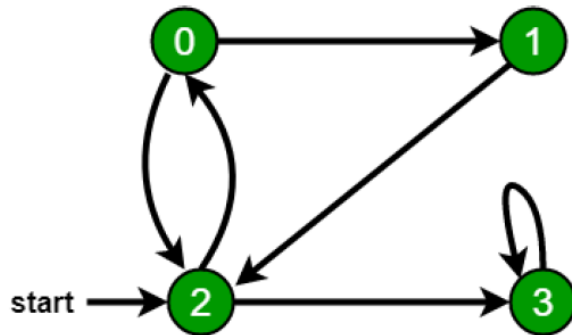
Muhammad Umair (12093)

Submitted To:

Sir Faiq

TASK

TASK: Consider the below graph;



Using DFS, check if there is any path exists between any two nodes? Also the return the path.

e.g. If user two vertices i.e. 2 and 1; the program should return : Yes the paths exist, which are [2,1],[2,0,1].

Code:

```
vertexList = ['0', '1', '2', '3']
edgeList = [(0, 1), (0, 2), (1, 2), (2, 0), (2, 3), (3, 3)]
graphs = (vertexList, edgeList)
```

```
def dfs(graph, start):
    vertexList, edgeList = graph
    visitedVertex = []
    stack = [start]
    adjacencyList = [[] for vertex in vertexList]

    for edge in edgeList:
        adjacencyList[edge[0]].append(edge[1])

    while stack:
        current = stack.pop()
        for neighbor in adjacencyList[current]:
            if not neighbor in visitedVertex:
                stack.append(neighbor)
        visitedVertex.append(current)
    return visitedVertex
```

```

def print_paths(s, d):
    if (s, d) in edgeList:
        print('yes the path exists')
        print(s, d)
    elif (s, d) not in edgeList:
        vertexlist_copy = [int(i) for i in vertexList]
        for i in vertexlist_copy:
            if (s, vertexlist_copy[i]) in edgeList:
                if (vertexlist_copy[i], vertexlist_copy[i+1]) in edgeList:
                    print('The path exists')
                    print(s, vertexlist_copy[i], d)
                    break
        else:
            print('The path do not exists')

print(dfs(graphs, 2))
print('For checking path')
st = int(input('Enter 1st Vertex: '))
end = int(input('Enter 2nd Vertex: '))
print_paths(st, end)

```

Output:



```

Task x
"C:\Users\muham\PycharmProjects\Muhammad Umair_12093_Lab07\venv\Scripts\python.exe"
[2, 3, 3, 0, 1]
For checking path
Enter 1st Vertex: 2
Enter 2nd Vertex: 1
The path exists
2 0 1

Process finished with exit code 0
|

```