

**Artificial intelligence (CSC403)**

**Student name: Razan Abdulrahman Alrashed**

**ID: 443500087**

**Assignment 2**

**DFS**

**Using Python Class**

1

3

2

4

6

5

class Graph:

    def \_\_init\_\_(self):

        self.graph = {}

    def add\_node(self, node):

        if node not in self.graph:

            self.graph[node] = []

    def add\_edge(self, node1, node2):

        if node1 in self.graph and node2 in self.graph:

            self.graph[node1].append(node2)

            self.graph[node2].append(node1)

def DFS(visited, graph, node, target):

    if node not in visited:

        print(node, end=' ')

        visited.add(node)

        if node == target:

            return True

        for neighbor in graph[node]:

            if neighbor not in visited:

                if DFS(visited, graph, neighbor, target):

                    return True

    return False

graph=Graph()

graph.add\_node('1')

graph.add\_node('2')

graph.add\_node('3')

graph.add\_node('4')

graph.add\_node('5')

graph.add\_node('6')

graph.add\_edge('1','2')

graph.add\_edge('1','3')

graph.add\_edge('2','4')

graph.add\_edge('2','5')

graph.add\_edge('3','6')

visited = set()

DFS(visited,graph.graph,'1','5')



*Output*