```
1 import numpy as np
2 class Graph:
       def __init__(self,vertices):
 3
           self._adjMat = np.zeros((vertices, vertices))
 4
           self._vertices = vertices
 5
 6
           self._visited = [0] * vertices
 7
8
       def insert_edge(self,u,v,w=1):
9
           self.\_adjMat[u][v] = w
10
11
       def delete_edge(self,u,v):
           self.\_adjMat[u][v] = 0
12
13
14
       def get_edge(self,u,v):
15
           return self._adjMat[u][v]
16
       def vertices_count(self):
17
           return self._vertices
18
19
       def edge_count(self):
20
           count = 0
21
           for i in range(self._vertices):
22
               for j in range(self._vertices):
23
                    if not self._adjMat[i][j] == 0:
24
                        count += 1
25
26
           return count
27
       def indegree(self,u):
28
           count = 0;
29
           for i in range(self._vertices):
30
               if not self._adjMat[i][u] == 0:
31
                    count += 1
32
           return count
33
34
       def outdegree(self,u):
35
           count = 0;
36
           for i in range(self._vertices):
37
               if not self._adjMat[u][i] == 0:
38
39
                    count += 1
40
           return count
41
42
       def display(self):
43
           print(self._adjMat)
44
45
       def DFS(self, source):
           if self._visited[source] == 0:
46
47
               print(source, end=' - ')
```

```
Learning Data Structures & Algorithms in Python from Scratch - File - D:\MyPythonLab\dfs.py
                 self._visited[source] = 1
49
                 for j in range(self._vertices):
                     if self._adjMat[source][j] == 1 and self.
50
   _visited[j] == 0:
51
                          self.DFS(j)
52
53 G = Graph(7)
54 G.insert_edge(0,1)
55 G.insert_edge(0,5)
56 G.insert_edge(0,6)
57 G.insert_edge(1,0)
58 G.insert_edge(1,2)
59 G.insert_edge(1,5)
60 G.insert_edge(1,6)
61 G.insert_edge(2,3)
62 G.insert_edge(2,4)
63 G.insert_edge(2,6)
64 G.insert_edge(3,4)
65 G.insert_edge(4,2)
66 G.insert_edge(4,5)
67 G.insert_edge(5,2)
68 G.insert_edge(5,3)
69 G.insert_edge(6,3)
70 G.DFS(0)
71
72
73
```