

## AdjMatrixGraph.java

```

1 /*
2  * Written by JJ Shepherd
3  */
4 import java.util.*;
5 public class AdjMatrixGraph
6 {
7     public static final int DEF_VERT_COUNT = 10;
8     private double[][] adjMatrix;
9     private LinkedList<Integer> markedList;
10    private Queue<Integer> vQ;
11    public AdjMatrixGraph()
12    {
13        init(DEF_VERT_COUNT);
14    }
15    public AdjMatrixGraph(int size)
16    {
17        init(size);
18    }
19    private void init(int size)
20    {
21        if(size <= 0)
22        {
23            init(DEF_VERT_COUNT);
24            return;
25        }
26        adjMatrix = new double[size][size];
27        for(int i=0;i<adjMatrix.length;i++)
28            for(int j=0;j<adjMatrix[i].length;j++)
29                adjMatrix[i][j] = 0;
30        markedList = new LinkedList<Integer>();
31        vQ = new LinkedList<Integer>();
32    }
33    public void addEdge(int fromVertex, int toVertex, double weight)
34    {
35        if(!isValid(fromVertex) || !isValid(toVertex))
36            return;
37        adjMatrix[fromVertex][toVertex] = weight;
38    }
39    private boolean isValid(int index)
40    {
41        return index >= 0 && index < adjMatrix.length;
42    }
43    public void printAdjMatrix()
44    {
45        for(int i=0;i<adjMatrix.length;i++)
46        {
47            for(int j=0;j<adjMatrix[i].length;j++)
48            {
49                System.out.print(adjMatrix[i][j]+" ");
50            }
51            System.out.println();
52        }
53    }
54    public void printDFS()
55    {
56        markedList.clear();
57        printDFS(0);

```

AdjMatrixGraph.java

```
58     }
59     private void printDFS(int index)
60     {
61         System.out.println(index);
62         markedList.add(index);
63         for(int i=0;i<adjMatrix.length;i++)
64             if(adjMatrix[index][i] != 0.0 && !markedList.contains(i))
65                 printDFS(i);
66     }
67     public void printBFS()
68     {
69         markedList.clear();
70         vQ.clear();
71         markedList.add(0);
72         vQ.add(0);
73         System.out.println(0);
74         while(!vQ.isEmpty())
75         {
76             int currI = vQ.remove();
77             markedList.add(currI);
78             for(int i=0;i<adjMatrix.length;i++)
79                 if(!markedList.contains(i) && !vQ.contains(i) && adjMatrix[currI][i] != 0.0)
80                 {
81                     System.out.println(i);
82                     vQ.add(i);
83                 }
84         }
85     }
86 }
87
```