

```

1 package Group8.MainClass;
2 import java.util.*;
3 public class Main {
4
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         Class c = new Class();
8         System.out.print("Input number of node : ");
9         int n = sc.nextInt();
10        int[][] weight = new int[n][n];
11        String[] temp = new String[n];
12        String[] name = new String[n];
13        name[0] = "u0";
14        for(int y=1;y<n;y++){
15            name[y] = "v"+y;
16        }
17        for(int y=0;y<n;y++){
18            System.out.print("Node "+name[y]+" : ");
19            String node = sc.next();
20            temp = node.split(",");
21            for(int z=0;z<n;z++){
22                weight[y][z] = Integer.valueOf(temp[z]);
23            }
24            temp = null;
25        }
26        System.out.println("\n
-----");
27        int visit[] = new int[n];
28        visit[0] = 1;
29
30        int stack[] = new int[n];
31        stack[0] = 0;
32        int stackw[] = new int[n];
33        stackw[0] = 0;
34        int minw = -1;
35        int minad = 0;
36        int minadwint = 0;
37        int minadw[] = new int[n];
38        String path[] = new String[n];
39
40        System.out.printf("%8s", "l(u0)");
41        for(int y=1;y<name.length;y++){
42            System.out.printf("%15s", name[y]);
43        }
44        System.out.printf("%15s\n", "Add to S");
45        System.out.printf("%8s", "0");
46        for(int y=1;y<name.length;y++){
47            System.out.printf("%15s", "(INFI,-)");
48        }
49        System.out.printf("%15s\n", "u0");
50
51        for (int i = 0; i < n - 1; i++) {
52            System.out.printf("%8s", "");
53            int w[] = new int[n];
54            int wnode[] = new int[n];
55
56            for (int j = 1; j < n; j++) {
57                if (visit[j] == 1 && j != 0) {
58                    System.out.printf("%15s", "");
59                } else if (visit[j] == 0) {
60                    for (int k = 0; k < stack.length; k++) {
61                        if(i>0){
62                            if (weight[j][stack[k]] != -1) {
63                                w[k] = weight[j][stack[k]]+stackw[stack[k]];
64                            } else if (weight[j][stack[k]] == -1) {
65                                w[k] = -1;
66                            }
67                        }
68                    }
69                    if(i==0){
70                        for (int k = 0; k < stack.length; k++) {
71                            w[k] = weight[j][stack[k]];
72                        }
73                    }
74                    minw = -1;
75                    boolean chk = false;
76                    for (int l=0;l<w.length;l++) {
77                        if (w[l] > -1 && visit[j] == 0) {
78                            chk = true;
79                            minw = w[l];
80                        }
81                    }
82                }
83            }

```

```

84         if (chk) {
85             for (int l : w) {
86                 if (visit[j] == 0) {
87                     if (l < minw && l != -1) {
88                         minw = l;
89                     }
90                 }
91             }
92         }
93
94         wnode[j] = minw;
95         minadw[j] = c.findindex(minw, w);
96         if (wnode[j] == -1) {
97             System.out.printf("%15s", "(INFI,-)");
98         } else {
99             System.out.printf("%15s", "(" + wnode[j] + ", " + name[stack[minadw[j]]] + ")");
100         }
101     }
102 }
103
104 for (int q = 0; q < wnode.length; q++) {
105     if (wnode[q] != -1 && visit[q] == 0) {
106         c.minn(wnode[q]);
107     }
108 }
109 int minn = c.getminn();
110 for (int q = 0; q < wnode.length; q++) {
111     if (wnode[q] < c.getminn() && wnode[q] != -1 && visit[q] == 0) {
112         c.minn(wnode[q]);
113     }
114 }
115
116 minad = c.findindex(c.getminn(), wnode);
117 minadwint = minadw[minad];
118 visit[minad] = 1;
119 stack[i + 1] = minad;
120 System.out.printf("%15s ", name[minad]);
121
122 stackw[minad] = stackw[stack[minadwint]] + weight[minad][stack[minadwint]];
123 path[i] = name[stack[minadwint]] + "->" + name[minad];
124 System.out.println("");
125 }
126
127 System.out.print("\nPath : ");
128 for (int t = 0; t < path.length - 1; t++) {
129     System.out.print(path[t] + " , ");
130 }
131 System.out.println("\b\b\b");
132 System.out.println(
133     "-----");
134 }
135

```

```
1 package Group8.MainClass;
2
3 public class Class {
4     private int minn;
5     private int minad;
6     private int min;
7
8     public void min(int min){
9
10         this.min = min;
11     }
12
13     public int getmin(){
14
15         return min;
16     }
17
18     public void minn(int minn){
19
20         this.minn = minn;
21     }
22
23     public int getminn(){
24
25         return minn;
26     }
27
28     public void minad(int minad){
29
30         this.minad = minad;
31     }
32
33     public int getminad(){
34
35         return minn;
36     }
37
38     public int findindex(int m,int[] n){
39         int index = 0;
40         int i = 0;
41         boolean chk = false;
42         for(i=0;i<n.length;i++){
43             if(n[i] == m) {
44                 index = i;
45                 chk = true;
46             }
47             if(chk) break;
48         }
49         return index;
50     }
51 }
52
```