

Student: Matthew Smyth

Project Due Date: 10/30/2020

Algorithm Steps:

```
step 0: open inFile, SSSfile, debugFile
numNodes get from inFile
Allocate and initialize all members in the DijkstraSSS class accordingly
step 1: loadCostMatrix (inFile)
sourceNode 1
step 2: setBestCostAry (sourceNode)
setFatherAry (...)
setMarkedAry (sourceNode)
step 3: minNode findMinNode(...)
markedAry[minNode] 1
debugPrint (...)
step 4: // expanding the minNode
currentNode 1
step 5: if markedAry[currentNode] == 0
newCost computeCost(minNode, currentNode)
if newCost < bestCostAry [currentNode]
bestCostAry[currentNode] newCost
fatherAry[currentNode] minNode

debugPrint (...)
Step 6: currentNode ++
Step 7: repeat step 5 to step 6 while currentNode <= numNodes
step 8: repeat step 3 to step 7 until all nodes are marked
// begin printing the paths
step 9: currentNode 1
step 10: printShortestPath (currentNode, sourceNode, SSSfile)
step 11: currentNode ++
step 12: repeat 10 and step 11 while currentNode <= numNodes
step 13: sourceNode ++
step 14: repeat step 2 to step 13 while sourceNode <= numNodes
step 15: close all files
```

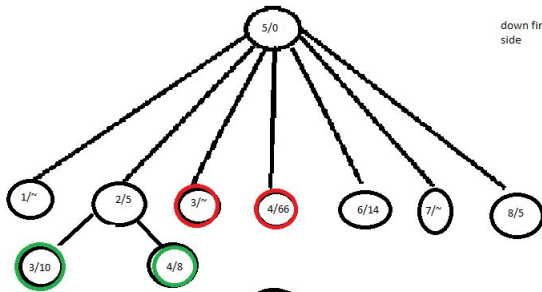
Cost Matrix

	1	2	3	4	5	6	7	8
1	0	30	5	19	29	99999	99999	99999
2	99999	0	5	3	99999	17	99999	2
3	99999	7	0	5	99999	99999	28	9
4	6	99999	33	0	8	3	99999	99999
5	99999	5	99999	66	0	14	99999	5
6	6	99999	6	24	99999	0	9	99999
7	99999	15	4	4	99999	3	0	99999
8	6	99999	99999	99999	2	7	32	0

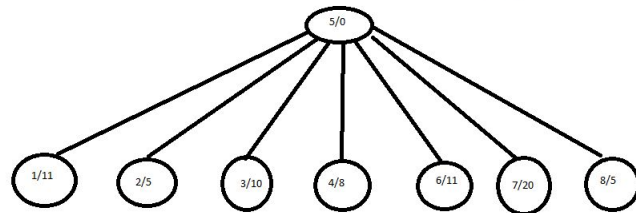
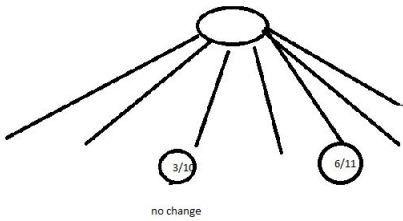
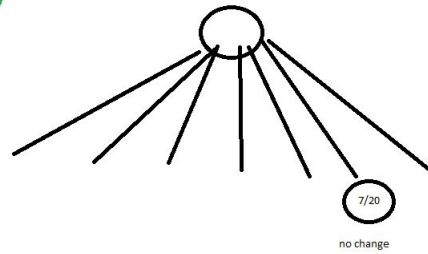
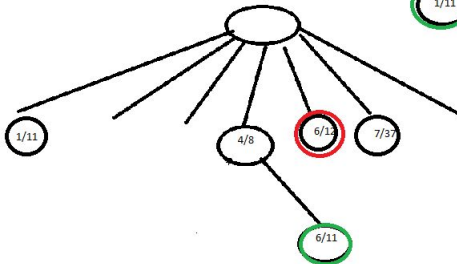
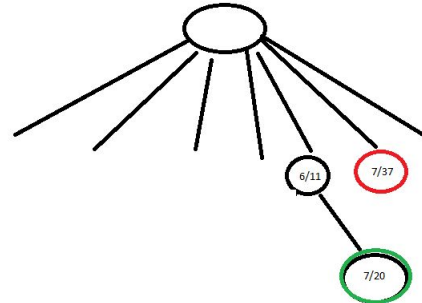
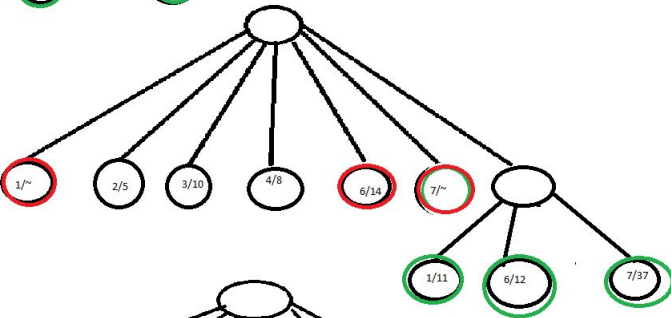
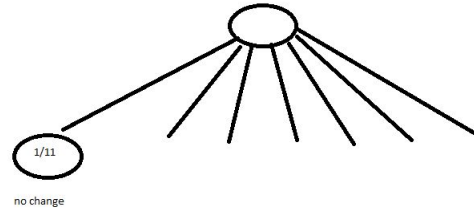
Source node 5:

fatherAry	5	5	5	5	5	5	5	5
bestCostAry	99999	5	99999	66	0	14	99999	5
markedAry	0	1	0	0	1	0	0	0
fatherAry	5	5	2	2	5	5	5	5
bestCostAry	99999	5	10	8	0	14	99999	5
markedAry	0	1	0	0	1	0	0	1
fatherAry	8	5	2	2	5	8	8	5
bestCostAry	11	5	10	8	0	12	37	5
markedAry	0	1	0	1	1	0	0	1
fatherAry	8	5	2	2	5	4	8	5
bestCostAry	11	5	10	8	0	11	37	5
markedAry	0	1	1	1	1	0	0	1

fatherAry	8	5	2	2	5	4	8	5
bestCostAry	11	5	10	8	0	11	37	5
markedAry	1	1	1	1	1	0	0	1
fatherAry	8	5	2	2	5	4	8	5
bestCostAry	11	5	10	8	0	11	37	5
markedAry	1	1	1	1	1	1	0	1
fatherAry	8	5	2	2	5	4	6	5
bestCostAry	11	5	10	8	0	11	20	5
markedAry	1	1	1	1	1	1	1	1



down first then right
side



```
#include <string>
#include <iostream>
#include <fstream>
```

```
using namespace std;
```

```
class DijkstraSSS {
public:
    int numNodes;
    int sourceNode = 1;
    int minNode = 0;
    int currentNode = 1;
    int newCost = 99999;
```

```

int **costMatrix;
int *fatherAry;
int *markedAry;
int *bestCostAry;

DijkstraSSS(int n) {
    numNodes = n;

    costMatrix = new int*[numNodes + 1];
    for (int i = 1; i < numNodes + 1; i++) {
        costMatrix[i] = new int[numNodes + 1];
    }
    for (int i = 1; i < numNodes + 1; i++) {
        for (int j = 1; j < numNodes + 1; j++) {
            if (i == j)
                costMatrix[i][i] = 0;
            else {
                costMatrix[i][j] = 99999;
            }
        }
    }

    fatherAry = new int[numNodes + 1];
    for (int i = 1; i < numNodes + 1; i++) {
        fatherAry[i] = i;
    }

    markedAry = new int[numNodes + 1];
    for (int i = 1; i < numNodes + 1; i++) {
        markedAry[i] = 0;
    }

    bestCostAry = new int[numNodes + 1];
    for (int i = 1; i < numNodes + 1; i++) {
        bestCostAry[i] = 99999;
    }
}

void loadCostMatrix(ifstream &inFile) {
    int node1;
    int node2;
    int cost;
    while (inFile) {
        inFile >> node1;

```

```

        inFile >> node2;
        inFile >> cost;
        costMatrix[node1][node2] = cost;
    }
}

void setBestCostAry() {
    for (int i = 1; i < numNodes + 1; i++) {
        bestCostAry[i] = costMatrix[sourceNode][i];
    }
}

void setFatherAry() {
    for (int i = 1; i < numNodes + 1; i++) {
        fatherAry[i] = sourceNode;
    }
}

void setMarkedAry() {
    for (int i = 1; i < numNodes + 1; i++) {
        markedAry[i] = 0;
    }
    markedAry[sourceNode] = 1;
}

int findMinNode() {
    int minCost = 99999;
    minNode = 0;
    int index = 1;
    while (index <= numNodes) {
        if (markedAry[index] == 0) {
            if (bestCostAry[index] < minCost) {
                minCost = bestCostAry[index];
                minNode = index;
            }
        }
        index++;
    }
    return minNode;
}

int computeCost() {
    int sum;
    sum = bestCostAry[minNode] + costMatrix[minNode][currentNode];
}

```

```

        return sum;
    }

    void debugPrint(ofstream &debugFile) {
        debugFile << "======" <<
endl;
        debugFile << "The sourceNode is: " << sourceNode << endl;

        debugFile << "The fatherAry is: " << endl;
        for (int i = 1; i < numNodes + 1; i++) {
            debugFile << fatherAry[i] << "\t";
        }

        debugFile << endl << "The bestCostAry is: " << endl;
        for (int i = 1; i < numNodes + 1; i++) {
            debugFile << bestCostAry[i] << "\t";
        }

        debugFile << endl << "The markedAry is: " << endl;
        for (int i = 1; i < numNodes + 1; i++) {
            debugFile << markedAry[i] << "\t";
        }

        debugFile << endl;
    }

    void printShortestPath(ofstream &SSSfile) {
        if (sourceNode == 1 && currentNode == 1) {
            SSSfile << "======" << endl;
            SSSfile << "There are " << numNodes
                << " nodes in the input graph. Below are all pairs of the
shortest paths:"
                << endl;
        }
        if (currentNode == 1) {
            SSSfile << "======" << endl;
            SSSfile << "The source node = " << sourceNode << endl << endl;
        }
        SSSfile << "The path from " << sourceNode << " to " << currentNode
            << ": ";
        SSSfile << currentNode << "<-";
        int path = fatherAry[currentNode];
        int cost = bestCostAry[currentNode];
        while (path != sourceNode) {

```



```

        SSSfile << path << "<-";
        cost += bestCostAry[path];
        path = fatherAry[path];
    }
    SSSfile << sourceNode;
    SSSfile << ": cost = " << cost << endl;
}
}
;

```

```

int main(int argc, const char *argv[]) {
    string in = argv[1];
    ifstream inFile = ifstream(in);
    string out = argv[2];
    ofstream SSSfile = ofstream(out);
    string out2 = argv[3];
    ofstream debugFile = ofstream(out2);

    int numNodes;
    inFile >> numNodes;
    DijkstraSSS sss = DijkstraSSS(numNodes);

    sss.loadCostMatrix(inFile);
    sss.sourceNode = 1;

    while (sss.sourceNode <= numNodes) {
        sss.setBestCostAry();
        sss.setFatherAry();
        sss.setMarkedAry();

        int flag = 1;
        while (flag) {
            sss.minNode = sss.findMinNode();
            sss.markedAry[sss.minNode] = 1;
            sss.debugPrint(debugFile);

            sss.currentNode = 1;

            while (sss.currentNode <= numNodes) {
                if (sss.markedAry[sss.currentNode] == 0) {
                    sss.newCost = sss.computeCost();
                    if (sss.newCost < sss.bestCostAry[sss.currentNode]) {
                        sss.bestCostAry[sss.currentNode] = sss.newCost;
                        sss.fatherAry[sss.currentNode] = sss.minNode;
                    }
                }
            }
        }
    }
}

```

```

        sss.debugPrint(debugFile);
    }
}
    sss.currentNode++;
}

    flag = 0;
    for (int i = 1; i < numNodes + 1; i++) {
        if (sss.markedAry[i] == 0) {
            flag = 1;
        }
    }
}

    sss.currentNode = 1;

    while (sss.currentNode <= numNodes) {
        sss.printShortestPath(SSSfile);

        sss.currentNode++;
    }

    sss.sourceNode++;
}

    inFile.close();
    debugFile.close();
    SSSfile.close();
    return 0;
}

```

Debugfile

=====

The sourceNode is: 1

The fatherAry is:

1 1 1 1 1 1 1 1
The bestCostAry is:
0 30 5 19 29 99999 99999 99999

The markedAry is:
1 0 1 0 0 0 0 0
=====

The sourceNode is: 1
The fatherAry is:
1 3 1 1 1 1 1 1

The bestCostAry is:
0 12 5 19 29 99999 99999 99999
The markedAry is:
1 0 1 0 0 0 0 0

=====

The sourceNode is: 1
The fatherAry is:
1 3 1 3 1 1 1 1

The bestCostAry is:
0 12 5 10 29 99999 99999 99999
The markedAry is:
1 0 1 0 0 0 0 0

=====

The sourceNode is: 1
The fatherAry is:
1 3 1 3 1 1 3 1

The bestCostAry is:
0 12 5 10 29 99999 33 99999
The markedAry is:
1 0 1 0 0 0 0 0

=====

The sourceNode is: 1
The fatherAry is:
1 3 1 3 1 1 3 3

The bestCostAry is:
0 12 5 10 29 99999 33 14
The markedAry is:
1 0 1 0 0 0 0 0

=====

The sourceNode is: 1
The fatherAry is:
1 3 1 3 1 1 3 3

The bestCostAry is:
0 12 5 10 29 99999 33 14
The markedAry is:

1 0 1 1 0 0 0 0
=====

The sourceNode is: 1

The fatherAry is:

1 3 1 3 4 1 3 3

The bestCostAry is:

0 12 5 10 18 99999 33 14

The markedAry is:

1 0 1 1 0 0 0 0
=====

The sourceNode is: 1

The fatherAry is:

1 3 1 3 4 4 3 3

The bestCostAry is:

0 12 5 10 18 13 33 14

The markedAry is:

1 0 1 1 0 0 0 0
=====

The sourceNode is: 1

The fatherAry is:

1 3 1 3 4 4 3 3

The bestCostAry is:

0 12 5 10 18 13 33 14

The markedAry is:

1 1 1 1 0 0 0 0
=====

The sourceNode is: 1

The fatherAry is:

1 3 1 3 4 4 3 3

The bestCostAry is:

0 12 5 10 18 13 33 14

The markedAry is:

1 1 1 1 0 1 0 0
=====

The sourceNode is: 1

The fatherAry is:

1 3 1 3 4 4 6 3

The bestCostAry is:

0 12 5 10 18 13 22 14

The markedAry is:

1 1 1 1 0 1 0 0
=====

The sourceNode is: 1

The fatherAry is:

```

1      3      1      3      4      4      6      3
The bestCostAry is:
0      12      5      10      18      13      22      14
The markedAry is:
1      1      1      1      0      1      0      1
=====
The sourceNode is: 1
The fatherAry is:
1      3      1      3      8      4      6      3
The bestCostAry is:
0      12      5      10      16      13      22      14
The markedAry is:
1      1      1      1      0      1      0      1
=====
The sourceNode is: 1
The fatherAry is:
1      3      1      3      8      4      6      3
The bestCostAry is:
0      12      5      10      16      13      22      14
The markedAry is:
1      1      1      1      1      1      0      1
=====
The sourceNode is: 1
The fatherAry is:
1      3      1      3      8      4      6      3
The bestCostAry is:
0      12      5      10      16      13      22      14
The markedAry is:
1      1      1      1      1      1      1      1
=====
The sourceNode is: 2
The fatherAry is:
2      2      2      2      2      2      2      2
The bestCostAry is:
99999 0      5      3      99999 17      99999 2
The markedAry is:
0      1      0      0      0      0      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      2      2      2      2
The bestCostAry is:
8      0      5      3      99999 17      99999 2
The markedAry is:

```

0 1 0 0 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 2 2 2

The bestCostAry is:

8 0 5 3 4 17 99999 2

The markedAry is:

0 1 0 0 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 8 2 2

The bestCostAry is:

8 0 5 3 4 9 99999 2

The markedAry is:

0 1 0 0 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 8 8 2

The bestCostAry is:

8 0 5 3 4 9 34 2

The markedAry is:

0 1 0 0 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 8 8 2

The bestCostAry is:

8 0 5 3 4 9 34 2

The markedAry is:

0 1 0 1 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 4 8 2

The bestCostAry is:

8 0 5 3 4 6 34 2

The markedAry is:

0 1 0 1 0 0 0 1
=====

The sourceNode is: 2

The fatherAry is:

```

8      2      2      2      8      4      8      2
The bestCostAry is:
8      0      5      3      4      6      34      2
The markedAry is:
0      1      0      1      1      0      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      8      4      8      2
The bestCostAry is:
8      0      5      3      4      6      34      2
The markedAry is:
0      1      1      1      1      0      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      8      4      3      2
The bestCostAry is:
8      0      5      3      4      6      33      2
The markedAry is:
0      1      1      1      1      0      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      8      4      3      2
The bestCostAry is:
8      0      5      3      4      6      33      2
The markedAry is:
0      1      1      1      1      1      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      8      4      6      2
The bestCostAry is:
8      0      5      3      4      6      15      2
The markedAry is:
0      1      1      1      1      1      0      1
=====
The sourceNode is: 2
The fatherAry is:
8      2      2      2      8      4      6      2
The bestCostAry is:
8      0      5      3      4      6      15      2
The markedAry is:

```

1 1 1 1 1 1 0 1
=====

The sourceNode is: 2

The fatherAry is:

8 2 2 2 8 4 6 2

The bestCostAry is:

8 0 5 3 4 6 15 2

The markedAry is:

1 1 1 1 1 1 1 1
=====

The sourceNode is: 3

The fatherAry is:

3 3 3 3 3 3 3 3

The bestCostAry is:

99999 7 0 5 99999 99999 28 9

The markedAry is:

0 0 1 1 0 0 0 0
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 3 3 3 3

The bestCostAry is:

11 7 0 5 99999 99999 28 9

The markedAry is:

0 0 1 1 0 0 0 0
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 4 3 3 3

The bestCostAry is:

11 7 0 5 13 99999 28 9

The markedAry is:

0 0 1 1 0 0 0 0
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 4 4 3 3

The bestCostAry is:

11 7 0 5 13 8 28 9

The markedAry is:

0 0 1 1 0 0 0 0
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 4 4 3 3
The bestCostAry is:
11 7 0 5 13 8 28 9
The markedAry is:
0 1 1 1 0 0 0 0
=====

The sourceNode is: 3
The fatherAry is:
4 3 3 3 4 4 3 3
The bestCostAry is:
11 7 0 5 13 8 28 9
The markedAry is:
0 1 1 1 0 1 0 0
=====

The sourceNode is: 3
The fatherAry is:
4 3 3 3 4 4 6 3
The bestCostAry is:
11 7 0 5 13 8 17 9
The markedAry is:
0 1 1 1 0 1 0 0
=====

The sourceNode is: 3
The fatherAry is:
4 3 3 3 4 4 6 3
The bestCostAry is:
11 7 0 5 13 8 17 9
The markedAry is:
0 1 1 1 0 1 0 1
=====

The sourceNode is: 3
The fatherAry is:
4 3 3 3 8 4 6 3
The bestCostAry is:
11 7 0 5 11 8 17 9
The markedAry is:
0 1 1 1 0 1 0 1
=====

The sourceNode is: 3
The fatherAry is:
4 3 3 3 8 4 6 3
The bestCostAry is:
11 7 0 5 11 8 17 9
The markedAry is:

1 1 1 1 0 1 0 1
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 8 4 6 3

The bestCostAry is:

11 7 0 5 11 8 17 9

The markedAry is:

1 1 1 1 1 1 0 1
=====

The sourceNode is: 3

The fatherAry is:

4 3 3 3 8 4 6 3

The bestCostAry is:

11 7 0 5 11 8 17 9

The markedAry is:

1 1 1 1 1 1 1 1
=====

The sourceNode is: 4

The fatherAry is:

4 4 4 4 4 4 4 4

The bestCostAry is:

6 99999 33 0 8 3 99999 99999

The markedAry is:

0 0 0 1 0 1 0 0
=====

The sourceNode is: 4

The fatherAry is:

4 4 6 4 4 4 4 4

The bestCostAry is:

6 99999 9 0 8 3 99999 99999

The markedAry is:

0 0 0 1 0 1 0 0
=====

The sourceNode is: 4

The fatherAry is:

4 4 6 4 4 4 6 4

The bestCostAry is:

6 99999 9 0 8 3 12 99999

The markedAry is:

0 0 0 1 0 1 0 0
=====

The sourceNode is: 4

The fatherAry is:

4 4 6 4 4 4 6 4
The bestCostAry is:
6 99999 9 0 8 3 12 99999

The markedAry is:
1 0 0 1 0 1 0 0
=====

The sourceNode is: 4
The fatherAry is:
4 1 6 4 4 4 6 4

The bestCostAry is:
6 36 9 0 8 3 12 99999
The markedAry is:

1 0 0 1 0 1 0 0
=====

The sourceNode is: 4
The fatherAry is:
4 1 6 4 4 4 6 4

The bestCostAry is:
6 36 9 0 8 3 12 99999
The markedAry is:

1 0 0 1 1 1 0 0
=====

The sourceNode is: 4
The fatherAry is:
4 5 6 4 4 4 6 4

The bestCostAry is:
6 13 9 0 8 3 12 99999
The markedAry is:

1 0 0 1 1 1 0 0
=====

The sourceNode is: 4
The fatherAry is:
4 5 6 4 4 4 6 5

The bestCostAry is:
6 13 9 0 8 3 12 13
The markedAry is:

1 0 0 1 1 1 0 0
=====

The sourceNode is: 4
The fatherAry is:
4 5 6 4 4 4 6 5

The bestCostAry is:
6 13 9 0 8 3 12 13
The markedAry is:

1 0 1 1 1 1 0 0
=====

The sourceNode is: 4

The fatherAry is:

4 5 6 4 4 4 6 5

The bestCostAry is:

6 13 9 0 8 3 12 13

The markedAry is:

1 0 1 1 1 1 1 0
=====

The sourceNode is: 4

The fatherAry is:

4 5 6 4 4 4 6 5

The bestCostAry is:

6 13 9 0 8 3 12 13

The markedAry is:

1 1 1 1 1 1 1 0
=====

The sourceNode is: 4

The fatherAry is:

4 5 6 4 4 4 6 5

The bestCostAry is:

6 13 9 0 8 3 12 13

The markedAry is:

1 1 1 1 1 1 1 1
=====

The sourceNode is: 5

The fatherAry is:

5 5 5 5 5 5 5 5

The bestCostAry is:

99999 5 99999 66 0 14 99999 5

The markedAry is:

0 1 0 0 1 0 0 0
=====

The sourceNode is: 5

The fatherAry is:

5 5 2 5 5 5 5 5

The bestCostAry is:

99999 5 10 66 0 14 99999 5

The markedAry is:

0 1 0 0 1 0 0 0
=====

The sourceNode is: 5

The fatherAry is:

```

5      5      2      2      5      5      5      5
The bestCostAry is:
99999 5      10     8      0      14     99999 5
The markedAry is:
0      1      0      0      1      0      0      0
=====
The sourceNode is: 5
The fatherAry is:
5      5      2      2      5      5      5      5
The bestCostAry is:
99999 5      10     8      0      14     99999 5
The markedAry is:
0      1      0      0      1      0      0      1
=====
The sourceNode is: 5
The fatherAry is:
8      5      2      2      5      5      5      5
The bestCostAry is:
11     5      10     8      0      14     99999 5
The markedAry is:
0      1      0      0      1      0      0      1
=====
The sourceNode is: 5
The fatherAry is:
8      5      2      2      5      8      5      5
The bestCostAry is:
11     5      10     8      0      12     99999 5
The markedAry is:
0      1      0      0      1      0      0      1
=====
The sourceNode is: 5
The fatherAry is:
8      5      2      2      5      8      8      5
The bestCostAry is:
11     5      10     8      0      12     37     5
The markedAry is:
0      1      0      0      1      0      0      1
=====
The sourceNode is: 5
The fatherAry is:
8      5      2      2      5      8      8      5
The bestCostAry is:
11     5      10     8      0      12     37     5
The markedAry is:

```

0 1 0 1 1 0 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 8 5

The bestCostAry is:

11 5 10 8 0 11 37 5

The markedAry is:

0 1 0 1 1 0 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 8 5

The bestCostAry is:

11 5 10 8 0 11 37 5

The markedAry is:

0 1 1 1 1 0 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 8 5

The bestCostAry is:

11 5 10 8 0 11 37 5

The markedAry is:

1 1 1 1 1 0 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 8 5

The bestCostAry is:

11 5 10 8 0 11 37 5

The markedAry is:

1 1 1 1 1 1 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 6 5

The bestCostAry is:

11 5 10 8 0 11 20 5

The markedAry is:

1 1 1 1 1 1 0 1
=====

The sourceNode is: 5

The fatherAry is:

8 5 2 2 5 4 6 5
The bestCostAry is:
11 5 10 8 0 11 20 5
The markedAry is:
1 1 1 1 1 1 1 1

=====
The sourceNode is: 6
The fatherAry is:
6 6 6 6 6 6 6 6
The bestCostAry is:
6 99999 6 24 99999 0 9 99999
The markedAry is:
1 0 0 0 0 1 0 0

=====
The sourceNode is: 6
The fatherAry is:
6 1 6 6 6 6 6 6
The bestCostAry is:
6 36 6 24 99999 0 9 99999
The markedAry is:
1 0 0 0 0 1 0 0

=====
The sourceNode is: 6
The fatherAry is:
6 1 6 6 1 6 6 6
The bestCostAry is:
6 36 6 24 35 0 9 99999
The markedAry is:
1 0 0 0 0 1 0 0

=====
The sourceNode is: 6
The fatherAry is:
6 1 6 6 1 6 6 6
The bestCostAry is:
6 36 6 24 35 0 9 99999
The markedAry is:
1 0 1 0 0 1 0 0

=====
The sourceNode is: 6
The fatherAry is:
6 3 6 6 1 6 6 6
The bestCostAry is:
6 13 6 24 35 0 9 99999
The markedAry is:

1 0 1 0 0 1 0 0
=====

The sourceNode is: 6

The fatherAry is:

6 3 6 3 1 6 6 6

The bestCostAry is:

6 13 6 11 35 0 9 99999

The markedAry is:

1 0 1 0 0 1 0 0
=====

The sourceNode is: 6

The fatherAry is:

6 3 6 3 1 6 6 3

The bestCostAry is:

6 13 6 11 35 0 9 15

The markedAry is:

1 0 1 0 0 1 0 0
=====

The sourceNode is: 6

The fatherAry is:

6 3 6 3 1 6 6 3

The bestCostAry is:

6 13 6 11 35 0 9 15

The markedAry is:

1 0 1 0 0 1 1 0
=====

The sourceNode is: 6

The fatherAry is:

6 3 6 3 1 6 6 3

The bestCostAry is:

6 13 6 11 35 0 9 15

The markedAry is:

1 0 1 1 0 1 1 0
=====

The sourceNode is: 6

The fatherAry is:

6 3 6 3 4 6 6 3

The bestCostAry is:

6 13 6 11 19 0 9 15

The markedAry is:

1 0 1 1 0 1 1 0
=====

The sourceNode is: 6

The fatherAry is:


```

6      3      6      3      4      6      6      3
The bestCostAry is:
6      13      6      11      19      0      9      15
The markedAry is:
1      1      1      1      0      1      1      0
=====

```

```

The sourceNode is: 6
The fatherAry is:
6      3      6      3      4      6      6      3
The bestCostAry is:
6      13      6      11      19      0      9      15
The markedAry is:
1      1      1      1      0      1      1      1
=====

```

```

The sourceNode is: 6
The fatherAry is:
6      3      6      3      8      6      6      3
The bestCostAry is:
6      13      6      11      17      0      9      15
The markedAry is:
1      1      1      1      0      1      1      1
=====

```

```

The sourceNode is: 6
The fatherAry is:
6      3      6      3      8      6      6      3
The bestCostAry is:
6      13      6      11      17      0      9      15
The markedAry is:
1      1      1      1      1      1      1      1
=====

```

```

The sourceNode is: 7
The fatherAry is:
7      7      7      7      7      7      7      7
The bestCostAry is:
99999 15      4      4      99999 3      0      99999
The markedAry is:
0      0      0      0      0      1      1      0
=====

```

```

The sourceNode is: 7
The fatherAry is:
6      7      7      7      7      7      7      7
The bestCostAry is:
9      15      4      4      99999 3      0      99999
The markedAry is:

```

0 0 0 0 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

6 7 7 7 7 7 7 7

The bestCostAry is:

9 15 4 4 99999 3 0 99999

The markedAry is:

0 0 1 0 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

6 3 7 7 7 7 7 7

The bestCostAry is:

9 11 4 4 99999 3 0 99999

The markedAry is:

0 0 1 0 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

6 3 7 7 7 7 7 3

The bestCostAry is:

9 11 4 4 99999 3 0 13

The markedAry is:

0 0 1 0 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

6 3 7 7 7 7 7 3

The bestCostAry is:

9 11 4 4 99999 3 0 13

The markedAry is:

0 0 1 1 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

6 3 7 7 4 7 7 3

The bestCostAry is:

9 11 4 4 12 3 0 13

The markedAry is:

0 0 1 1 0 1 1 0
=====

The sourceNode is: 7

The fatherAry is:

```

6      3      7      7      4      7      7      3
The bestCostAry is:
9      11     4      4      12     3      0      13
The markedAry is:
1      0      1      1      0      1      1      0
=====
The sourceNode is: 7
The fatherAry is:
6      3      7      7      4      7      7      3
The bestCostAry is:
9      11     4      4      12     3      0      13
The markedAry is:
1      1      1      1      0      1      1      0
=====
The sourceNode is: 7
The fatherAry is:
6      3      7      7      4      7      7      3
The bestCostAry is:
9      11     4      4      12     3      0      13
The markedAry is:
1      1      1      1      1      1      1      0
=====
The sourceNode is: 7
The fatherAry is:
6      3      7      7      4      7      7      3
The bestCostAry is:
9      11     4      4      12     3      0      13
The markedAry is:
1      1      1      1      1      1      1      1
=====
The sourceNode is: 8
The fatherAry is:
8      8      8      8      8      8      8      8
The bestCostAry is:
6      99999 99999 99999 2      7      32     0
The markedAry is:
0      0      0      0      1      0      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      8      8      8      8      8      8
The bestCostAry is:
6      7      99999 99999 2      7      32     0
The markedAry is:

```

0 0 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

8 5 8 5 8 8 8 8

The bestCostAry is:

6 7 99999 68 2 7 32 0

The markedAry is:

0 0 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

8 5 8 5 8 8 8 8

The bestCostAry is:

6 7 99999 68 2 7 32 0

The markedAry is:

1 0 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

8 5 1 5 8 8 8 8

The bestCostAry is:

6 7 11 68 2 7 32 0

The markedAry is:

1 0 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

8 5 1 1 8 8 8 8

The bestCostAry is:

6 7 11 25 2 7 32 0

The markedAry is:

1 0 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

8 5 1 1 8 8 8 8

The bestCostAry is:

6 7 11 25 2 7 32 0

The markedAry is:

1 1 0 0 1 0 0 1
=====

The sourceNode is: 8

The fatherAry is:

```

8      5      1      2      8      8      8      8
The bestCostAry is:
6      7      11     10     2      7      32     0
The markedAry is:
1      1      0      0      1      0      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      1      2      8      8      8      8
The bestCostAry is:
6      7      11     10     2      7      32     0
The markedAry is:
1      1      0      0      1      1      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      1      2      8      8      6      8
The bestCostAry is:
6      7      11     10     2      7      16     0
The markedAry is:
1      1      0      0      1      1      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      1      2      8      8      6      8
The bestCostAry is:
6      7      11     10     2      7      16     0
The markedAry is:
1      1      0      1      1      1      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      1      2      8      8      6      8
The bestCostAry is:
6      7      11     10     2      7      16     0
The markedAry is:
1      1      1      1      1      1      0      1
=====
The sourceNode is: 8
The fatherAry is:
8      5      1      2      8      8      6      8
The bestCostAry is:
6      7      11     10     2      7      16     0
The markedAry is:

```

1 1 1 1 1 1 1 1

SSSfile

=====

There are 8 nodes in the input graph. Below are all pairs of the shortest paths:

=====

The source node = 1

The path from 1 to 1: 1<-1: cost = 0

The path from 1 to 2: 2<-3<-1: cost = 17

The path from 1 to 3: 3<-1: cost = 5

The path from 1 to 4: 4<-3<-1: cost = 15

The path from 1 to 5: 5<-8<-3<-1: cost = 35

The path from 1 to 6: 6<-4<-3<-1: cost = 28

The path from 1 to 7: 7<-6<-4<-3<-1: cost = 50

The path from 1 to 8: 8<-3<-1: cost = 19

=====

The source node = 2

The path from 2 to 1: 1<-8<-2: cost = 10

The path from 2 to 2: 2<-2: cost = 0

The path from 2 to 3: 3<-2: cost = 5

The path from 2 to 4: 4<-2: cost = 3

The path from 2 to 5: 5<-8<-2: cost = 6

The path from 2 to 6: 6<-4<-2: cost = 9

The path from 2 to 7: 7<-6<-4<-2: cost = 24

The path from 2 to 8: 8<-2: cost = 2

=====

The source node = 3

The path from 3 to 1: 1<-4<-3: cost = 16

The path from 3 to 2: 2<-3: cost = 7

The path from 3 to 3: 3<-3: cost = 0

The path from 3 to 4: 4<-3: cost = 5

The path from 3 to 5: 5<-8<-3: cost = 20

The path from 3 to 6: 6<-4<-3: cost = 13

The path from 3 to 7: 7<-6<-4<-3: cost = 30

The path from 3 to 8: 8<-3: cost = 9

=====

The source node = 4

The path from 4 to 1: 1<-4: cost = 6

The path from 4 to 2: 2<-5<-4: cost = 21

The path from 4 to 3: 3<-6<-4: cost = 12

The path from 4 to 4: 4<-4: cost = 0

The path from 4 to 5: 5<-4: cost = 8

The path from 4 to 6: 6<-4: cost = 3

The path from 4 to 7: 7<-6<-4: cost = 15

The path from 4 to 8: 8<-5<-4: cost = 21

=====

The source node = 5

The path from 5 to 1: 1<-8<-5: cost = 16

The path from 5 to 2: 2<-5: cost = 5

The path from 5 to 3: 3<-2<-5: cost = 15

The path from 5 to 4: 4<-2<-5: cost = 13

The path from 5 to 5: 5<-5: cost = 0

The path from 5 to 6: 6<-4<-2<-5: cost = 24

The path from 5 to 7: 7<-6<-4<-2<-5: cost = 44

The path from 5 to 8: 8<-5: cost = 5

=====

The source node = 6

The path from 6 to 1: 1<-6: cost = 6

The path from 6 to 2: 2<-3<-6: cost = 19

The path from 6 to 3: 3<-6: cost = 6

The path from 6 to 4: 4<-3<-6: cost = 17

The path from 6 to 5: 5<-8<-3<-6: cost = 38

The path from 6 to 6: 6<-6: cost = 0

The path from 6 to 7: 7<-6: cost = 9

The path from 6 to 8: 8<-3<-6: cost = 21

=====

The source node = 7

The path from 7 to 1: 1<-6<-7: cost = 12

The path from 7 to 2: 2<-3<-7: cost = 15

The path from 7 to 3: 3<-7: cost = 4

The path from 7 to 4: 4<-7: cost = 4

The path from 7 to 5: 5<-4<-7: cost = 16

The path from 7 to 6: 6<-7: cost = 3

The path from 7 to 7: 7<-7: cost = 0

The path from 7 to 8: 8<-3<-7: cost = 17

=====

The source node = 8

The path from 8 to 1: 1<-8: cost = 6

The path from 8 to 2: 2<-5<-8: cost = 9

The path from 8 to 3: 3<-1<-8: cost = 17

The path from 8 to 4: 4<-2<-5<-8: cost = 19

The path from 8 to 5: 5<-8: cost = 2

The path from 8 to 6: 6<-8: cost = 7

The path from 8 to 7: 7<-6<-8: cost = 23

The path from 8 to 8: 8<-8: cost = 0