**package** GraphTheory;

**import** java.util.Scanner;

**class** DijstraAlgorithm

{

**public int**[][] **graph**;

**public int vertex**;

**public int edge**;

**public int**[] **distance**;

**public boolean**[] **visited**;

Scanner **input**;

**public** DijstraAlgorithm(**int**[][] graph, **int** vertex, **int** edge) {

**this**.**graph** = graph;

**this**.**vertex** = vertex;

**this**.**edge** = edge;

**this**.**distance** = **new int**[vertex];

**this**.**visited** = **new boolean**[vertex];

**this**.**input** = **new** Scanner(System.***in***);

}

**public void** Dijstra()

{

**for** (**int** i=0;i<**vertex**;i++)

{

**distance**[i] = Integer.***MAX\_VALUE***;

**visited**[i] = **false**;

}

**distance**[0] = 0;

**for** (**int** i=0;i<**edge**;i++)

{

**int** n1 = **input**.nextInt();

**int** n2 = **input**.nextInt();

**int** weight = **input**.nextInt();

**graph**[n1][n2] = weight;

**graph**[n2][n1] = weight;

}

**for** (**int** i=0;i<**vertex**-1;i++)

{

**int** minVertex = *findMinVertex*(**distance**,**visited**,**vertex**);

**visited**[minVertex] = **true**;

**for** (**int** j=0;j<**vertex**;j++)

{

**if** (!**visited**[j] && **graph**[minVertex][j] !=0 && **distance**[minVertex] != Integer.***MAX\_VALUE***)

{

**int** newDistance = **distance**[minVertex] + **graph**[minVertex][j];

**if** (newDistance<**distance**[j])

{

**distance**[j] = newDistance;

}

}

}

}

**for** (**int** i=0;i<**vertex**;i++)

{

System.***out***.println(**"Distance "**+i+**": "**+**distance**[i]);

}

}

**public static int** findMinVertex(**int**[] distance,**boolean**[] visited,**int** vertex)

{

**int** minVertex = Integer.***MAX\_VALUE***;

**for** (**int** i=0;i<vertex;i++)

{

**if** (!visited[i] && minVertex>distance[i])

{

minVertex = i;

}

}

**return** minVertex;

}

}

**public class** DijstraAlgorithmUndirected {

**public static void** main(String[] args) {

**int**[][] graph = **new int**[100][100];

Scanner input = **new** Scanner(System.***in***);

System.***out***.println(**"Enter number of vertex: "**);

**int** vertex = input.nextInt();

System.***out***.println(**"Enter number of edges: "**);

**int** edge = input.nextInt();

DijstraAlgorithm object = **new** DijstraAlgorithm(graph,vertex,edge);

object.Dijstra();

}

}