public class Graph{

public static void main(Strings [] args){

class Graph{

int [][] graph;

int vertices;

boolean isdirected;

Graph.addEdge(src:0,dest:1);

graph.addEdge(src:0,dest:2);

graph.addEdge(src:0,dest:3);

graph.addEdge(src:1,dest:2);

graph.printGraph();

}

Graph(int vertices,boolean isdirected){

graph=new int[vertices][vertices];

this.vertices=vertices;

this.isdirected=isdirected;

}

void removeEdge(int src,int dest){

if(isvaild(src,dest)){

graph[src][dest]=0;

if(!isdirected)

graph[dest][src]=0;

}else{

System.err.println(x:"invaild source and destination.");

}

}

void addEdge(int src,int dest){

if(isvaild(src,dest)){

graph[src][dest]=1;

if(!isdirected){

graph[src][dest]=1;

}

}else{

System.err.println("invaild source and destination.");

}

}

boolean isvaild(int src,int dest){

return src>=0 && dest>=0 && src<vertices && dest<vertices && src!=dest;

}

void printGraph(){

System.out.println("matrix representation of Graph");

for(int i=0;i<vertices;i++){

for(int j=0;j<vertices;j++){

System.out.print(graph[i][j]+" ");

}

System.out.print();

}

}

}

}