# Graph Representation using Adjacency Matrix

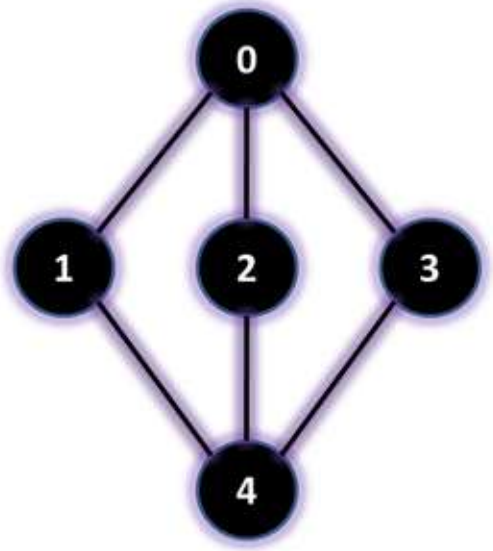


**You are given a graph in terms of its number of vertices and edges. Your task is to construct its adjacency matrix representation.

The graph can be directed or undirected, and each edge has a weight.e a program to implement a graph using an adjacency Matrix.**

** - first2 inputs are number of vertices and edges
- Then graph is directed or un directed followed by edges.**

# Adjacency Matrix




| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 |
| 2 | 1 | 0 | 0 | 0 | 1 |
| 3 | 1 | 0 | 0 | 0 | 1 |
| 4 | 0 | 1 | 1 | 1 | 0 |

**Input Format**

First line:
An integer V — the number of vertices in the graph.
Second line:
An integer E — the number of edges.
Third line:
A string — "yes" if the graph is directed, "no" if the graph is undirected.
Next E lines:
Each line contains three integers: u v w, denoting an edge from node u to node v with weight w.

5
6
no
0 1 1
1 4 1

2 4 1
0 2 1
3 4 1
0 3 1

### Constraints

1 ≤ V ≤ 100
0 ≤ E ≤ V*(V-1)
0 ≤ u, v < V
0 ≤ w ≤ 100
The graph may have self-loops (i.e., u == v is allowed).

### Output Format

Print the V x V adjacency matrix.
Each row should contain V space-separated integers.
If there is no edge between two vertices, the value should be 0.

0 1 1 1 0
1 0 0 0 1
1 0 0 0 1
1 0 0 0 1
0 1 1 1 0

### Sample Input 0

```
5
6
no
0 1 1
1 4 1
2 4 1
0 2 1
```

```
3 4 1
0 3 1
```

### Sample Output 0

```
0 1 1 1 0
1 0 0 0 1
1 0 0 0 1
1 0 0 0 1
0 1 1 1 0
```

### Explanation 0

The graph is undirected, so every edge u v w implies both matrix[u][v] = w and matrix[v][u] = w.

The matrix represents edge weights; 0 means no edge.

For instance, the edge 0 1 1 creates both matrix[0][1] = 1 and matrix[1][0] = 1.

```java
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc=new Scanner(System.in);
        int v=sc.nextInt();
        int e=sc.nextInt();
        int st,ed,w;
        String str=sc.next();
        int gra[][]=new int[v][v];
        for(int i=0;i<e;i++){
            st=sc.nextInt();
            ed=sc.nextInt();
            w=sc.nextInt();
            gra[st][ed]=w;
            if(str.equals("no")){
                gra[ed][st]=w;
            }
        }
        for(int i=0;i<v;i++){
            for(int j=0;j<v;j++){
                System.out.print(gra[i][j]+" ");
            }
            System.out.println();
        }
    }
}
```

Line: 1 Col: 1

Upload Code as File  Test against custom input  Run Code  Submit Code