

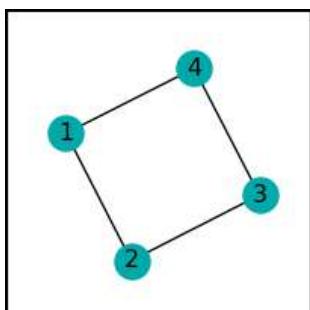
### A. Matriz de Adjacência

time limit per test: 1 second

memory limit per test: 256 megabytes

Existem diversas formas diferentes de se representar um grafo. Uma delas é a matriz de adjacência. Nela, cada linha e cada coluna representam um vértice do grafo. Se o vértice da linha  $i$  está conectado ao vértice da coluna  $j$ , então o valor da matriz na posição  $(i, j)$  é 1. Caso contrário, o valor é 0.

Por exemplo, o grafo não-direcionado abaixo



pode ser representado pela seguinte matriz de adjacência:

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

Neste exercício, sua tarefa é, dado uma lista de conexões de um grafo, representá-lo como uma matriz de adjacência.

#### Input

A entrada é composta por um único caso de testes. A primeira linha da entrada contém dois inteiros  $V$  ( $2 \leq V \leq 100$ ) e  $C$  ( $0 \leq C \leq 10000$ ), que representam o número de vértices do grafo e o número de conexões entre vértices do grafo. As próximas  $C$  linhas contêm dois inteiros  $A$  e  $B$  ( $1 \leq A, B \leq V$ ), indicando que os vértices  $A$  e  $B$  estão conectados.

#### Output

A saída deve conter uma matriz de adjacência do grafo, com  $N$  linhas e  $N$  colunas, com os valores separados por um espaço.

#### Examples

<b>input</b>	<input type="button" value="Copy"/>
4 4	
1 2	
2 3	
3 4	
4 1	
<b>output</b>	<input type="button" value="Copy"/>
0 1 0 1	
1 0 1 0	
0 1 0 1	
1 0 1 0	
<b>input</b>	<input type="button" value="Copy"/>
5 3	
1 2	

#### IDP - TAA - 2025/02

Private

Participant



#### → About Group



Este grupo tem o objetivo de organizar as atividades de programação da disciplina de Técnicas de Programação e Análise de Algoritmos.

[Group website](#)

#### → Group Contests

- TAA - LEE 06
- TAA - LEE 05
- TAA - LEE 05
- TAA - LEE 04
- TAA - LEE 04
- TAA - AS 01
- TAA - LEA 03
- TAA - LE 03
- TAA - LEA 02
- TAA - LEE 02
- TAA - LEA 01
- TAA - LEE 01
- ET - Exercício de Testes

#### TAA - LEE 06

Contest is running

3 days

Contestant



#### → Submit?

1

2 4  
5 1**output**

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

Language: GNU G++17 7.3.0

Choose file:

**input**

```
4 6
4 1
4 2
3 4
2 3
1 2
1 3
```

**output**

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

[Codeforces](#) (c) Copyright 2010-2025 Mike Mirzayanov

The only programming contests Web 2.0 platform

Server time: Nov/13/2025 23:19:58<sup>UTC-3</sup> (j1).

Desktop version, switch to [mobile version](#).

[Privacy Policy](#) | [Terms and Conditions](#)

Supported by



ITMO