

## E. Flawed Flow

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Emuskald considers himself a master of flow algorithms. Now he has completed his most ingenious program yet — it calculates the maximum flow in an undirected graph. The graph consists of  $n$  vertices and  $m$  edges. Vertices are numbered from 1 to  $n$ . Vertices 1 and  $n$  being the source and the sink respectively.

However, his max-flow algorithm seems to have a little flaw — it only finds the flow volume for each edge, but not its direction. Help him find for each edge the direction of the flow through this edges. Note, that the resulting flow should be correct maximum flow.

More formally. You are given an undirected graph. For each it's undirected edge  $(a_i, b_i)$  you are given the flow volume  $c_i$ . You should direct all edges in such way that the following conditions hold:

- for each vertex  $v$  ( $1 < v < n$ ), sum of  $c_i$  of incoming edges is equal to the sum of  $c_i$  of outgoing edges;
- vertex with number 1 has no incoming edges;
- the obtained directed graph **does not have cycles**.

### Input

The first line of input contains two space-separated integers  $n$  and  $m$  ( $2 \leq n \leq 2 \cdot 10^5$ ,  $n - 1 \leq m \leq 2 \cdot 10^5$ ), the number of vertices and edges in the graph. The following  $m$  lines contain three space-separated integers  $a_i$ ,  $b_i$  and  $c_i$  ( $1 \leq a_i, b_i \leq n$ ,  $a_i \neq b_i$ ,  $1 \leq c_i \leq 10^4$ ), which means that there is an undirected edge from  $a_i$  to  $b_i$  with flow volume  $c_i$ .

It is guaranteed that there are no two edges connecting the same vertices; the given graph is connected; a solution always exists.

### Output

Output  $m$  lines, each containing one integer  $d_i$ , which should be 0 if the direction of the  $i$ -th edge is  $a_i \rightarrow b_i$  (the flow goes from vertex  $a_i$  to vertex  $b_i$ ) and should be 1 otherwise. The edges are numbered from 1 to  $m$  in the order they are given in the input.

If there are several solutions you can print any of them.

### Examples

input	Copy
3 3 3 2 10 1 2 10 3 1 5	
output	Copy
1 0 1	

### Codeforces Round #165 (Div. 2)

**Finished**

#### → Practice?

Want to solve the contest problems after the official contest ends? Just register for practice and you will be able to submit solutions.

[Register for practice](#)

#### → Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

[Start virtual contest](#)

#### → Problem tags

[dfs and similar](#) [sortings](#) [\\*2100](#)

No tag edit access

#### → Contest materials

- Announcement #1 (ru) ×
- Announcement #2 ×
- Tutorial (ru) ×

input	Copy
<pre> 4 5 1 2 10 1 3 10 2 3 5 4 2 15 3 4 5 </pre>	
output	Copy
<pre> 0 0 1 1 0 </pre>	

### Note

In the first test case, 10 flow units pass through path  $1 \rightarrow 2 \rightarrow 3$ , and 5 flow units pass directly from source to sink:  $1 \rightarrow 3$ .

---

[Codeforces](#) (c) Copyright 2010-2020 Mike Mirzayanov  
 The only programming contests Web 2.0 platform  
 Server time: Sep/22/2020 15:40:14<sup>UTC-5</sup> (i1).  
 Desktop version, switch to [mobile version](#).  
[Privacy Policy](#)

Supported by

