

A. Thor

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Thor is getting used to the Earth. As a gift Loki gave him a smartphone. There are n applications on this phone. Thor is fascinated by this phone. He has only one minor issue: he can't count the number of unread notifications generated by those applications (maybe Loki put a curse on it so he can't).

q events are about to happen (in chronological order). They are of three types:

1. Application x generates a notification (this new notification is unread).
2. Thor reads all notifications generated so far by application x (he may re-read some notifications).
3. Thor reads the first t notifications generated by phone applications (notifications generated in first t events of the first type). It's guaranteed that there were at least t events of the first type before this event. Please note that he doesn't read first t unread notifications, he just reads the very first t notifications generated on his phone and he may re-read some of them in this operation.

Please help Thor and tell him the number of unread notifications after each event. You may assume that initially there are no notifications in the phone.

Input

The first line of input contains two integers n and q ($1 \leq n, q \leq 300\,000$) — the number of applications and the number of events to happen.

The next q lines contain the events. The i -th of these lines starts with an integer $type_i$ — type of the i -th event. If $type_i = 1$ or $type_i = 2$ then it is followed by an integer x_i . Otherwise it is followed by an integer t_i ($1 \leq type_i \leq 3$, $1 \leq x_i \leq n$, $1 \leq t_i \leq q$).

Output

Print the number of unread notifications after each event.

Examples

| input | Copy |
|---------------------------------|------|
| 3 4 1 3 1 1 1 2 2 3 | |
| output | Copy |
| 1 2 3 | |

Codeforces Round #366 (Div. 1)

Finished

→ 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

brute force data structures
implementation *1600

No tag edit access

→ Contest materials

- Announcement (en) 
- Tutorial (en) 

2

input

Copy

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

output

Copy

```
1
2
3
0
1
2
```

Note

In the first sample:

1. Application 3 generates a notification (there is 1 unread notification).
2. Application 1 generates a notification (there are 2 unread notifications).
3. Application 2 generates a notification (there are 3 unread notifications).
4. Thor reads the notification generated by application 3, there are 2 unread notifications left.

In the second sample test:

1. Application 2 generates a notification (there is 1 unread notification).
2. Application 4 generates a notification (there are 2 unread notifications).
3. Application 2 generates a notification (there are 3 unread notifications).
4. Thor reads first three notifications and since there are only three of them so far, there will be no unread notification left.
5. Application 3 generates a notification (there is 1 unread notification).
6. Application 3 generates a notification (there are 2 unread notifications).

[Codeforces](#) (c) Copyright 2010-2020 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Jun/09/2020 18:59:46^{UTC-4} (i1).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY