Metro

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	1024 MB

You just got inside a metro, and you want to take a seat somewhere. Obviously, you don't want to sit next to anyone. As a matter of fact, you want to sit as far away as you can from anyone. Thinking about that while standing up, you think of the following social experiment.

Suppose you have a metro that has a row of $N + 2$ seats, numbered from 0 to $N + 1$. Seats 0 and $N + 1$ are already occupied. A normal person chooses a vacant seat such that it is as far away from other occupied seats. That is, for each unoccupied seat, the person calculates two values d_l and d_r, which are the distances to the nearest seat from the left and right, respectively. Out of each seat, the person will choose the seat with the value $min(d_l, d_r)$ as high as possible. If there are multiple such seats, the person will choose the seat with the value $max(d_l, d_r)$ as high as possible. If the person still can't decide on a seat, he will choose the leftmost seat (that is, the seat with the smallest index).

Now, let's start the social experiment. There are Q events that happen sequentially, numbered from 1 to Q. Each event is described by two numbers: t_i and k_i.

1. If t_i is 1, then a random person occupies seat k_i if that place is vacant. Otherwise, the person sitting there will stand up and leave the metro, freeing that seat.

2. If t_i is 2, then you wonder: if k_i normal people would get inside the metro, where would the last person to enter sit?

Note: for the events with $t_i = 2$, the configuration of the seats will not be changed by the k_i people, as this is just a hypothetical question.

Input

The first line of input contains the integers N ($1 \leq N \leq 1.000.000.000$) and Q ($1 \leq Q \leq 100.000$). The following Q lines contain the description of each event, the i-th line containing two numbers t_i ($1 \leq t_i \leq 2$) and k_i ($1 \leq k_i \leq N$).

It is guaranteed that there are at most 11.000 events with $t_i = 1$.

Output

For each event of type 2, type on one line the index of the seat chosen by the last person. If the last person cannot find a seat, print -1 instead.

Example

standard input	standard output
10 5	6
2 5	1
1 4	7
2 5	-1
2 1	
2 10	