Friendly Tournament (`weights`)

Next year is our year!

The village of Krkva eventually decided to organize a football tournament. After aggressive advertisements, online campaigns, and various other promotions, they managed to lure N teams ready to fight for what could be their only piece of silverware for years to come.

However, the mayor wants to have an unusual tournament and to make it as exciting as possible, they want to calculate certain indicators to further improve their analysis.

Figure 1: A very intense match between two teams of different strengths.

Thus, the organizers initially assumed that the N teams have the same relative strength, which starts at 1. Then, as the tournament progresses, there will be Q events, given in chronological order, that are of one of two types:

- 1 a b: Two teams with relative strengths a and b play against each other. The winner will have a relative strength equal to $a + b$ from now on, while the loser is eliminated from the tournament (thus, one less team remains in the tournament). It is guaranteed that there exists at least one team with strength a and at least one team with strength b (if $a = b$, then there exist at least two teams with that strength level).

- 2 k: The Krkvan Football Federation wants to compute all pairwise differences in relative strength among the remaining teams and find the kth smallest such difference, where k is between 1 and the total number of such pairs (initially, this number would be at most $\frac{N \cdot (N-1)}{2}$).

As an example, if the relative strengths of the teams are $[4, 3, 1, 1]$, then the pairwise differences are $[|4 - 3|, |4 - 1|, |4 - 1|, |3 - 1|, |3 - 1|, |1 - 1|] = [1, 3, 3, 2, 2, 0]$, or $[0, 1, 2, 2, 3, 3]$ when sorted in increasing order. If k is 4, then the answer is 2.

The tournament is coming soon, so Krkva calls for your help to find these answers, or otherwise, the teams will fail to win any silverware for another year.

> ☞ Among the attachments of this task you may find a template file `weights.*` with a sample incomplete implementation.

Input

The first line of the input contains N and Q, the number of teams and the number of events.

The next Q lines contain the description of the events, as mentioned in the statement.

Output

The output shall contain one line for each event of type 2, representing the kth smallest difference in that respective event.

Constraints

- $1 \leq N, Q \leq 300\,000$.
- For operations of type 2, $1 \leq K \leq \frac{x \cdot (x-1)}{2}$, where x is the number of teams still in the event at the given moment.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points)   Examples.

– **Subtask 2** (9 points) $1 \leq N, Q \leq 100$.

– **Subtask 3** (14 points) $1 \leq N, Q \leq 1000$.

– **Subtask 4** (12 points)   All events of type 1 take place before all events of type 2 and there will be at most 500 teams remaining after performing the events of type 1.

– **Subtask 5** (10 points)   All events of type 1 take place before all events of type 2.

– **Subtask 6** (21 points) $1 \leq N, Q \leq 50\,000$.

– **Subtask 7** (34 points)   No additional limitations.

Examples

input	output
10 15 2 45 1 1 1 1 1 1 1 1 1 2 9 2 10 1 1 2 2 1 2 5 2 13 1 1 1 1 2 2 2 1 2 4 2 6	0 0 1 0 1 2 1 2 3