

D. Hanger

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In one very large and very respectable company there is a cloakroom with a coat hanger. It is represented by n hooks, positioned in a row. The hooks are numbered with positive integers from 1 to n from the left to the right.

The company workers have a very complicated work schedule. At the beginning of a work day all the employees are not there and the coat hanger in the cloakroom is empty. At some moments of time the employees arrive and some of them leave.

When some employee arrives, he hangs his cloak on one of the available hooks. To be of as little discomfort to his colleagues as possible, the hook where the coat will hang, is chosen like this. First the employee chooses the longest segment among available hooks following in a row. If there are several of such segments, then he chooses the one closest to the right. After that the coat is hung on the hook located in the middle of this segment. If the segment has an even number of hooks, then among two central hooks we choose the one closest to the right.

When an employee leaves, he takes his coat. As all the company workers deeply respect each other, no one takes somebody else's coat.

From time to time the director of this respectable company gets bored and he sends his secretary to see how many coats hang on the coat hanger from the i -th to the j -th hook inclusive. And this whim is always to be fulfilled, otherwise the director gets angry and has a mental breakdown.

Not to spend too much time traversing from the director's office to the cloakroom and back again, the secretary asked you to write a program, emulating the company cloakroom's work.

Input

The first line contains two integers n, q ($1 \leq n \leq 10^9$, $1 \leq q \leq 10^5$), which are the number of hooks on the hanger and the number of requests correspondingly. Then follow q lines with requests, sorted according to time. The request of the type " $0\ i\ j$ " ($1 \leq i \leq j \leq n$) — is the director's request. The input data has at least one director's request. In all other cases the request contains a positive integer not exceeding 10^9 — an employee identifier. Each odd appearance of the identifier of an employee in the request list is his arrival. Each even one is his leaving. All employees have distinct identifiers. When any employee arrives, there is always at least one free hook.

Output

For each director's request in the input data print a single number on a single line — the number of coats hanging on the hooks from the i -th one to the j -th one inclusive.

Examples

input
9 11 1 2 0 5 8 1 1 3 0 3 8 9 0 6 9 6 0 1 9
output

2
3
2
5