Shaking Stack

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2.5 seconds
Memory limit:	256 megabytes

Soum is currently working on his newest creation — a stack.

Soum's creation is a very normal, usual stack (though don't tell Soum that, he likes to think his creation is a unique masterpiece).

Soum starts with an empty stack on a table. At each point of time, either Soum will place a book on his stack, or Leo (who doesn't want Soum to make the greatest stack of all time) will shake the table so that the topmost book falls off the stack. Leo will only shake the table when there is at least one book on the stack.

After all these events, Soum wonders, after how many book placements were a certain pair of books both on the stack? Soum asks himself this question for Q different pairs of books. Can you help Soum find the answer for each queried pair of books?

Input

The first line of input will contain two space-separated integers N and Q. N is the number of events on Soum's stack, while Q is the number of pairs he is looking to find the answer for.

The following N lines begin with an integer T_i.

If T_i is 1, the line will contain another integer x, and this means that Soum places book number x on his stack. Soum will never place a book onto a stack twice, even if Leo has shaken the book off the stack. Therefore, all values of x in a single testcase are distinct.

Otherwise, T_i is 2, and this means that Leo shakes the table, leading to the topmost book falling off the stack.

The following Q lines each contain two space-separated integers A and B, the pair of books Soum is looking to find the answer for.

Output

For each query of two books A and B, output a single integer L, which represents the number of times Soum places a book, and the stack directly afterward contains both books A and B.

Scoring

The input is divided into multiple subtasks. Each subtask contains a number of testcases. The score for a subtask is awarded if your code passes all the testcases within the subtask.

The score of your submission is the sum of the scores for all subtasks it solves correctly.

For all subtasks, $1 \leq N \leq 200,000$ and $1 \leq Q \leq 200,000$. Additionally, $1 \leq A, B \leq N$ and $A \neq B$. Please note that it is **not** necessary that A or B have been on the stack at even one point of time (if either A or B has never been on the stack, the answer is 0 as Soum will have never seen them on the stack together).

- Subtask 1 [4 points]: $N = 2$, $Q = 1$
- Subtask 2 [7 points]: $N \leq 100$, $Q \leq 100$
- Subtask 3 [6 points]: $N \leq 500$, $Q \leq 500$ and $T_i \neq 2$ (Leo doesn't shake the table)
- Subtask 4 [10 points]: $N \leq 500$, $Q \leq 500$
- Subtask 5 [12 points]: $N \leq 2000$, $Q = 1$

- Subtask 6 [13 points]: $N \leq 2000$, $Q \leq 2000$ and $T_i \neq 2$ (Leo doesn't shake the table)
- Subtask 7 [14 points]: $N \leq 2000$, $Q \leq 2000$
- Subtask 8 [9 points]: $N \leq 200,000$, $Q = 1$
- Subtask 9 [11 points]: $N \leq 200,000$, $Q \leq 200,000$ and $T_i \neq 2$ (Leo doesn't shake the table)
- Subtask 10 [14 points]: No additional constraints.

Example

standard input	standard output
8 5 1 4 1 5 1 2 2 2 1 6 2 1 8 1 3 8 4 4 2 6 8 5 4	0 1 1 0 2

Note

In the first sample testcase, there are 8 events and 5 queries.

At the beginning, Soum's stack is empty. It looks like this: [] from bottom to top.

- Then, Soum adds book number 4 to the stack. It looks like this: [4].
- Then, Soum adds book number 5 to the stack. It looks like this: [4, 5].
- Then, Soum adds book number 2 to the stack. It looks like this: [4, 5, 2].
- Then, Leo shakes the stack and the topmost book, 2, falls off. It now looks like this: [4, 5].
- Then, Leo shakes the stack and the topmost book, 5, falls off. It now looks like this: [4].
- Then, Soum adds book number 6 to the stack. It looks like this: [4, 6].
- Then, Leo shakes the stack and the topmost book, 6 falls off. It now looks like this: [4].
- Then, Soum adds book number 6 to the stack. It looks like this: [4, 8].

Soum has seen the stack in the following states:

- 1st time: [4]
- 2nd time: [4, 5]
- 3rd time: [4, 5, 2]
- 4th time: [4, 6]

- 5th time: $[4, 8]$

Out of these, he has seen $(8, 4)$ together on the stack only once (5th time). He has seen $(5, 4)$ on the stack twice (2nd time and 3rd time). He has seen $(4, 2)$ on the stack once (3rd time). He has not seen any other queried pair.