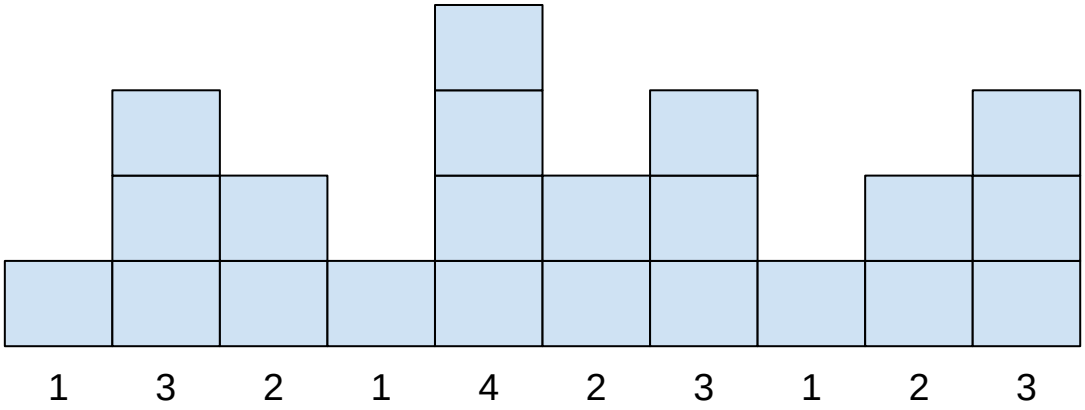
Problem H: Painting competition

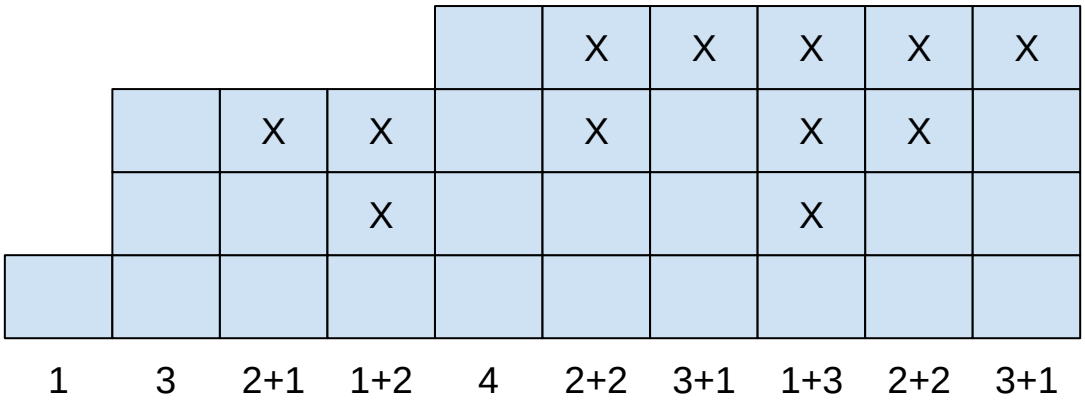
Time limit: 3s; Memory limit: 512 MB

You're joining a painting competition. The problem is drawing a line and controlling its stroke.

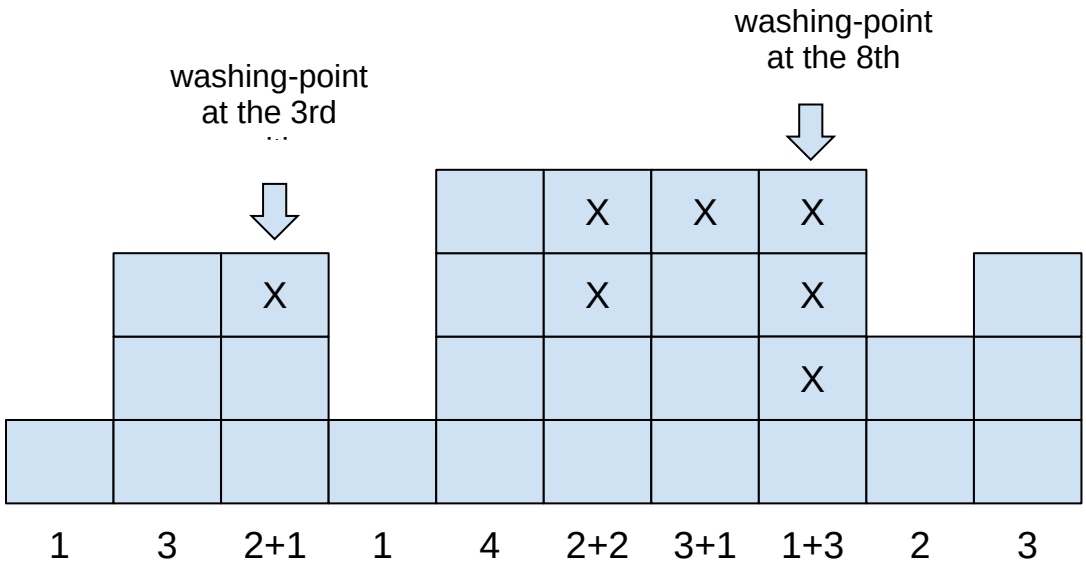
The problem can be described as an array of length n containing multiple integers a_i ($1 \leq i \leq n$) - expected stroke at position i-th. Contestants need to follow these integers to draw from left to right and control the line stroke. For example, if the exercise is an array $a = [1, 3, 2, 1, 4, 2, 3, 1, 2, 3]$ ($n = 10$), the expected line will be:



However, the paintbrush is too difficult to control. After increasing the line stroke, you could not decrease it until washing it. Then, the picture became (X pixels are caused by the paintbrush problem).



There are some moments in the competition, the contestants **have to** wash their paintbrush called **washing-point**. Afterward, you can restart the drawing with the required stroke from the next point. For example, if you wash the paintbrush at 3rd and 8th position (arrow symbols), the stroke will be reset immediately after those positions.



A sub-problem $a[l,r]$ is a contiguous subsequence $a_l\ a_{l+1}\ a_{l+2} \ldots . a_r$ of the original problem (supposed that the problem is only this subsequence and its **washing-point** at that moment, the remaining parts of the original problem will be ignored).

The sub-problem score $P(a[l,r])$ will be calculated by the number of incorrect pixels of the picture (number of X pixels).

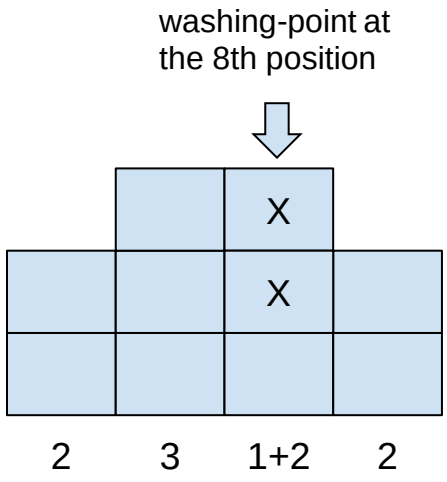
For example, if there is no **washing-point** in the original problem:

$$P(a[1,n]) = 1+2+2+1+3+2+1 = 12$$

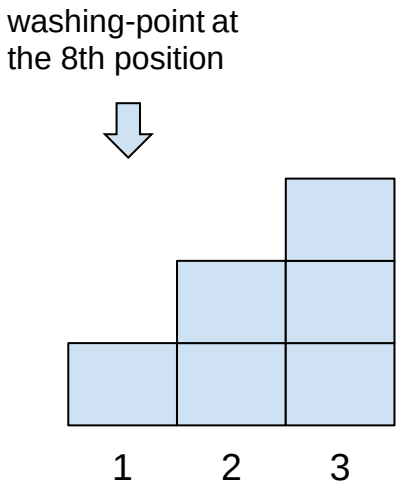
If there 2 **washing-point** at 3rd and 8th positions

$$P(a[1,n]) = 1 + 2 + 1 + 3 = 7$$

There are more examples for $a[6,9]$ and $a[8,10]$ (with **washing-points**).



P(a[6, 9]) = 2



P(a[8, 10]) = 0

However, the original problem is too long and boring. So the organizers decided to give contestants the following requests overtime (instead of completing the whole original problem):

- Request 1: Change the expected stroke at a specific position to a new value. In other words, given two numbers p and x, assign $a[p] = x$. This change will be retained.
- Request 2: Add a new **washing-point** at p. It guarantees that the position p is not an existing **washing-point**. This change will be retained.
- Request 3: Ask the contestant to complete the sub-problem $a[l, r]$.

Given the original problem and several requests. You need to answer the sub-problem score $P(a[l, r])$ that the contestant will get for each **request 3**.

Input

The first line contains two integers n and q ($1 \leq n, q \leq 2 \cdot 10^5$). The size of the given problem and the number of requests correspondingly.

The second line contains n integers a_i ($1 \leq i \leq n, 1 \leq a_i \leq 15$). The original problem description.

The next q lines describe requests. Each request will be one of following input formats:

- $1\ p\ x$: Change the expected stroke at p-th position to x ($1 \leq p \leq n, 1 \leq x \leq 15$).
- $2\ p$: Add a new **washing-point** at the p-th position ($1 \leq p \leq n$). It guarantees that the p-th position is not an existing **washing-point**. In other words, we will NOT have any two 2nd type requests with the same p value.
- $3\ l\ r$: Calculate the point $P(a[l,r])$ that the contestant will get supposed the sub-problem is $a[l,r]$ ($1 \leq l \leq r \leq n$). Then output the $P(a[l,r])$ value.

Output

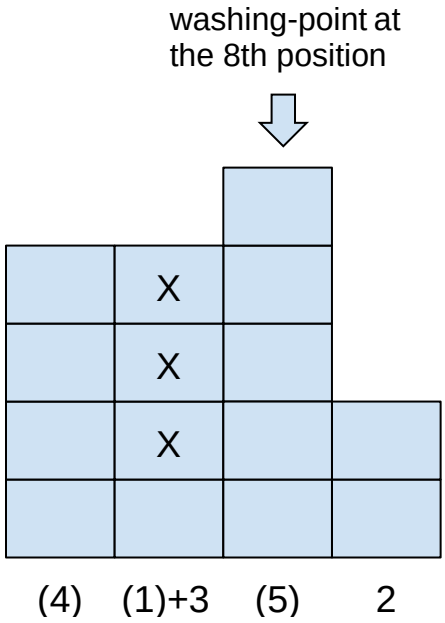
Each line of output will be an answer for the 3rd request.

Sample

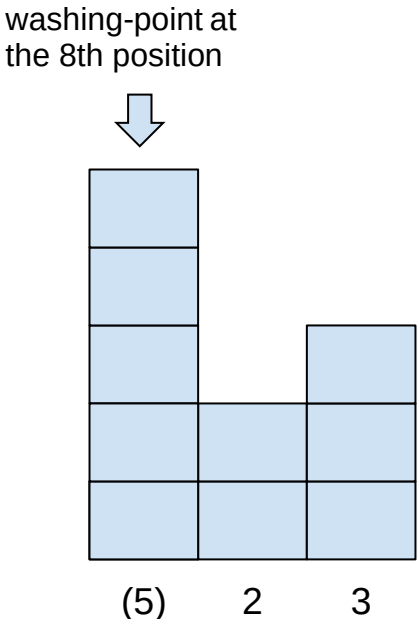
Input	Output
10 11 1 3 2 1 4 2 3 1 2 3 3 1 10 2 8 2 3 3 1 10 3 6 9 3 8 10 1 8 5 1 6 4 1 7 1 3 6 9 3 8 10	12 7 2 0 3 0
8 6 1 2 4 1 6 2 4 3 3 2 4 2 6 3 1 8 1 4 3 1 5 1 3 3 5	3 8 4

Explanation of the first case:

The 10th and 11th request will be - (x) value mean the expected stroke was changed by a previous request:



P(a[6, 9]) = 3



P(a[8, 10]) = 0