E. Boredom

Ilya is sitting in a waiting area of Metropolis airport and is bored of looking at time table that shows again and again that his plane is delayed. So he took out a sheet of paper and decided to solve some problems.

First Ilya has drawn a grid of size $n \times n$ and marked n squares on it, such that no two marked squares share the same row or the same column. He calls a rectangle on a grid with sides parallel to grid sides *beautiful* if exactly two of its corner squares are marked. There are exactly $n \cdot (n - 1) / 2$ beautiful rectangles.

Ilya has chosen q query rectangles on a grid with sides parallel to grid sides (not necessarily beautiful ones), and for each of those rectangles he wants to find its *beauty degree*. Beauty degree of a rectangle is the number of beautiful rectangles that share at least one square with the given one.

Now Ilya thinks that he might not have enough time to solve the problem till the departure of his flight. You are given the description of marked cells and the query rectangles, help Ilya find the beauty degree of each of the query rectangles.

Input

The first line of input contains two integers n and q ($2 \le n \le 200\,000$, $1 \le q \le 200\,000$) — the size of the grid and the number of query rectangles.

The second line contains n integers $p_1, p_2, ..., p_n$, separated by spaces ($1 \le p_i \le n$, all p_i are different), they specify grid squares marked by Ilya: in column i he has marked a square at row p_i, rows are numbered from 1 to n, bottom to top, columns are numbered from 1 to n, left to right.

The following q lines describe query rectangles. Each rectangle is described by four integers: l, d, r, u ($1 \le l \le r \le n$, $1 \le d \le u \le n$), here l and r are the leftmost and the rightmost columns of the rectangle, d and u the bottommost and the topmost rows of the rectangle.

Output

For each query rectangle output its beauty degree on a separate line.

Examples

input
2 3
1 2
1 1 1 1
1 1 1 2
1 1 2 2

output
1
1
1

input
4 2
1 3 2 4
4 1 4 4
1 1 2 3

output
3
5

Note

The first sample test has one beautiful rectangle that occupies the whole grid, therefore the answer to any query is 1.

In the second sample test the first query rectangle intersects 3 beautiful rectangles, as shown on the picture below:

There are 5 beautiful rectangles that intersect the second query rectangle, as shown on the following picture: