

## C. Foe Pairs

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

You are given a permutation  $p$  of length  $n$ . Also you are given  $m$  foe pairs  $(a_i, b_i)$  ( $1 \leq a_i, b_i \leq n, a_i \neq b_i$ ).

Your task is to count the number of different intervals  $(x, y)$  ( $1 \leq x \leq y \leq n$ ) that do not contain any foe pairs. So you shouldn't count intervals  $(x, y)$  that contain at least one foe pair in it (the positions and order of the values from the foe pair are not important).

Consider some example:  $p = [1, 3, 2, 4]$  and foe pairs are  $\{(3, 2), (4, 2)\}$ . The interval  $(1, 3)$  is incorrect because it contains a foe pair  $(3, 2)$ . The interval  $(1, 4)$  is also incorrect because it contains two foe pairs  $(3, 2)$  and  $(4, 2)$ . But the interval  $(1, 2)$  is correct because it doesn't contain any foe pair.

### Input

The first line contains two integers  $n$  and  $m$  ( $1 \leq n, m \leq 3 \cdot 10^5$ ) — the length of the permutation  $p$  and the number of foe pairs.

The second line contains  $n$  distinct integers  $p_i$  ( $1 \leq p_i \leq n$ ) — the elements of the permutation  $p$ .

Each of the next  $m$  lines contains two integers  $(a_i, b_i)$  ( $1 \leq a_i, b_i \leq n, a_i \neq b_i$ ) — the  $i$ -th foe pair. Note a foe pair can appear multiple times in the given list.

### Output

Print the only integer  $c$  — the number of different intervals  $(x, y)$  that does not contain any foe pairs.

Note that the answer can be too large, so you should use 64-bit integer type to store it. In C++ you can use the `long long` integer type and in Java you can use `long` integer type.

### Examples

| input                        |
|------------------------------|
| 4 2<br>1 3 2 4<br>3 2<br>2 4 |
| output                       |
| 5                            |

| input   |
|---|
| 9 5<br>9 7 2 3 1 4 6 5 8<br>1 6<br>4 5<br>2 7<br>7 2<br>2 7 |
| output  |
| 20  |

### Note

In the first example the intervals from the answer are  $(1, 1)$ ,  $(1, 2)$ ,  $(2, 2)$ ,  $(3, 3)$  and  $(4, 4)$ .