In python, we can use `len(start(a[L:R]))` to calculate the number of distinct values of elements $a[L]$, $a[L+1]$, $\ldots$, $a[R-1]$.

Here are some interactive examples that may help you understand how it is done. Remember that the indices of python lists start from 0.

```
>>>
a=[1,2,1,3,2,1,4]
>>> print a[1:6]
[2, 1, 3, 2, 1]
>>> print set(a[1:6])
set([1, 2, 3])
>>> print
len(set(a[1:6]))
3
>>> a[3]=2
>>> print
len(set(a[1:6]))
2
>>> print len(set(a[3:5]))
1
```

Your task is to simulate this process.

Input

There will be only one test case. The first line contains two integers n and m $(1 \le n, m \le 50,000)$. The next line contains the original list.

All the integers are between 1 and 1,000,000 (inclusive). The next m lines contain the statements that you need to execute.

A line formatted as '`M` x y' $(1 \le y \le 1,000,000)$ means "$a[x] = y$", and a line formatted as '`Q` x y' means "print len(set($a[x : y]$))".

It is guaranteed that the statements will not cause "index out of range" error.

Output

Print the simulated result, one line for each query.

Sample Input

```
7 4
1 2 1 3 2 1 4
Q 1 6
M 3 2
Q 1 6
Q 3 5
```

Sample Output

```
3
2
1
```