Task 3: `Snacks`

Shor the Duck has prepared n plates of snacks to enjoy while watching movies! The i-th plate initially contains a snack with a deliciousness value of $a[i]$.

You need to process q queries. In the j-th query, Shor will do **both** of the following, in order:

1. Eat every snack whose deliciousness is between $l[j]$ and $r[j]$ (inclusive).

2. Then, replace each eaten snack with a new snack of deliciousness $x[j]$.

Before processing any queries, and after each query, Shor wants you to determine the sum of deliciousness of snacks across all plates.

Formally, you are given an array a of length n and must process q queries. Before processing any queries, print the sum of all elements in a. In the j-th query, update every element $a[i]$ such that $l[j] \leq a[i] \leq r[j]$ by setting $a[i] = x[j]$, then print the updated sum of all elements in a.

Input format

Your program must read from standard input.

The first line of input contains two space-separated integers n and q.

The second line of input contains n space-separated integers $a[1], a[2], \ldots, a[n]$.

The following q lines of input each contain three space-separated integers. The j-th of these lines contains $l[j], r[j]$, and $x[j]$, describing the j-th query.

Output format

Your program must print to standard output.

The output should contain $q + 1$ lines.

The first line of output should contain a single integer, the sum of all elements in a before all queries.

The following q lines of input should each contain one integer. The i-th of these lines should contain the sum of elements in a after the i-th query.

Subtasks

For all test cases, the input will satisfy the following bounds:

- $1 \leq n \leq 200\,000$

- $0 \leq q \leq 200\,000$

- $0 \leq a[i] \leq 10^9$ for all $1 \leq i \leq n$

- $0 \leq x[j] \leq 10^9$ for all $1 \leq j \leq q$

- $0 \leq l[j] \leq r[j] \leq 10^9$ for all $1 \leq j \leq q$

Your program will be tested on input instances that satisfy the following restrictions:

Subtask	Marks	Additional Constraints
0	0	Sample test cases
1	5	$q = 0$
2	12	$n, q \leq 2000$
3	21	$l[j] = r[j] \leq 200\,000$ and $a[i], x[j] \leq 200\,000$
4	13	$l[j] = r[j]$
5	16	$x[j] = 0$
6	33	No additional constraints

Sample Test Case 1

This test case is valid for subtasks 2, 3, 4, and 6.

Input	Output
5 3 1 6 2 4 6 6 6 3 2 2 3 3 3 5	19 13 14 20

Sample Test Case 2

This test case is valid for subtasks 2, 5, and 6.

Input	Output
6 4 929 121 5 3 919 72 1 133 0 70 79 0 900 999 0 1 1000 0	2049 1848 1848 0 0

Sample Test Case 3

This test case is valid for subtasks 2 and 6.

Input	Output
6 5 7 72 727 123 321 9 7 9 10 10 72 727 111 222 30 123 727 99 111 222 333	1259 1263 3352 3259 525 525

Sample Test Case 3 Explanation

Before all queries, the array a is $[7, 72, 727, 123, 321, 9]$, with a sum of 1259.

After the first query, the array a becomes $[10, 72, 727, 123, 321, 10]$, with a sum of 1263.

After the second query, the array a becomes $[727, 727, 727, 123, 321, 727]$, with a sum of 3352.

After the third query, the array a becomes $[727, 727, 727, 30, 321, 727]$, with a sum of 3259.

After the fourth query, the array a becomes $[99, 99, 99, 30, 99, 99]$, with a sum of 525.

After the fifth query, the array a becomes $[99, 99, 99, 30, 99, 99]$ with a sum of 525.