

F. Leha and security system

time limit per test: 2 seconds

memory limit per test: 512 megabytes

input: standard input

output: standard output

Bankopolis, the city you already know, finally got a new bank opened! Unfortunately, its security system is not yet working fine... Meanwhile hacker Leha arrived in Bankopolis and decided to test the system!

Bank has n cells for clients' money. A sequence from n numbers a_1, a_2, \dots, a_n describes the amount of money each client has. Leha wants to make requests to the database of the bank, finding out the total amount of money on some subsegments of the sequence and changing values of the sequence on some subsegments. Using a bug in the system, Leha can requests two types of queries to the database:

- $1 \ l \ r \ x \ y$ denoting that Leha changes each digit x to digit y in each element of sequence a_i , for which $l \leq i \leq r$ is holds. For example, if we change in number 11984381 digit 8 to 4, we get 11944341. It's worth noting that Leha, in order to stay in the shadow, never changes digits in the database to 0, i.e. $y \neq 0$.
- $2 \ l \ r$ denoting that Leha asks to calculate and print the sum of such elements of sequence a_i , for which $l \leq i \leq r$ holds.

As Leha is a white-hat hacker, he don't want to test this vulnerability on a real database. You are to write a similar database for Leha to test.

Input

The first line of input contains two integers n and q ($1 \leq n \leq 10^5$, $1 \leq q \leq 10^5$) denoting amount of cells in the bank and total amount of queries respectively.

The following line contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i < 10^9$) denoting the amount of money in each cell initially. These integers do not contain leading zeros.

Each of the following q lines has one of the formats:

- $1 \ l \ r \ x \ y$ ($1 \leq l \leq r \leq n$, $0 \leq x \leq 9$, $1 \leq y \leq 9$), denoting Leha asks to change each digit x on digit y for each element a_i of the sequence for which $l \leq i \leq r$ holds;
- $2 \ l \ r$ ($1 \leq l \leq r \leq n$), denoting you have to calculate and print the sum of elements a_i for which $l \leq i \leq r$ holds.

Output

For each second type query print a single number denoting the required sum.

Examples

input
5 5 38 43 4 12 70 1 1 3 4 8 2 2 4 1 4 5 0 8 1 2 5 8 7 2 1 5
output
103 207

input
5 5 25 36 39 40 899

1 1 3 2 7
2 1 2
1 3 5 9 1
1 4 4 0 9
2 1 5
output
111
1002

Note

Let's look at the example testcase.

Initially the sequence is [38, 43, 4, 12, 70].

After the first change each digit equal to 4 becomes 8 for each element with index in interval [1; 3]. Thus, the new sequence is [38, 83, 8, 12, 70].

The answer for the first sum's query is the sum in the interval [2; 4], which equal $83 + 8 + 12 = 103$, so the answer to this query is 103.

The sequence becomes [38, 83, 8, 12, 78] after the second change and [38, 73, 7, 12, 77] after the third.

The answer for the second sum's query is $38 + 73 + 7 + 12 + 77 = 207$.