

Problem D. 1D Spreadsheet

Input file: 1d-spreadsheet.in
Output file: 1d-spreadsheet.out
Time limit: 10 seconds
Memory limit: 512 mebibytes

You've got a marvellous opportunity to work as an intern for a huge software company over summer. The project you are assigned to is a brand new, cutting-edge application — 1D spreadsheet.

A 1D spreadsheet is an array of n cells numbered from 1 to n . The cell i may contain either an integer a_i or a *link* to some other cell l_i (in which case we will say that the cell i is *linked* to the cell l_i). Cyclic link dependencies are not allowed, that is, at any moment, there is no sequence of cells i_1, i_2, \dots, i_k such that the cell i_1 is linked to the cell i_2, \dots , and the cell i_k is linked to the cell i_1 .

Every cell can be *evaluated* to produce its *value*, which is defined as follows: if the cell i contains a number a_i then $value(i) = a_i$, otherwise $value(i) = value(l_i)$; that is, to evaluate a cell one must follow a chain of links that starts at the cell until he arrives at a cell that contains a number. As cyclic dependencies are disallowed, at any moment, $value(i)$ can be determined unambiguously for every cell i .

Your colleague Bill has just finished developing a new feature for the spreadsheet — range value summation. With this feature enabled, user may ask queries of sort “find $\sum_{i=l}^r value(i)$ ” as well as change the contents of cells. However, after some performance testing it turned out that Bill's implementation is quite slow. Your task is to optimize this feature. Write a program that processes a sequence of queries to 1D spreadsheet.

Input

The first line of input contains two space-separated integers n and q ($1 \leq n, q \leq 2 \cdot 10^5$) — the number of cells in the spreadsheet and the number of queries to process, respectively.

The second line contains n space-separated integers a_1, \dots, a_n ($-10^4 \leq a_i \leq 10^4$) — the initial values of cells. Initially, no cell contains a link.

The next q lines contain descriptions of queries according to the following format:

- “set i x ” — assign the cell i to contain the number x ($-10^4 \leq x \leq 10^4$)
- “link i j ” — assign the cell i to contain link to the cell j .
- “sum l r ” — find $\sum_{i=l}^r value(i)$.

It is guaranteed that cyclic dependencies will not occur at any point of time.

Output

Output the answers for “sum l r ” queries in the same order as in the input, one per line.

Example

1d-spreadsheet.in	1d-spreadsheet.out
5 7	15
1 2 3 4 5	19
sum 1 5	1
link 3 5	
link 1 3	
link 4 2	
sum 1 5	
set 5 -1	
sum 1 5	