frubana

CONECTANDO EL CAMPO

Entregable

Puedes utilizar Javascript, Typescript, Kotlin o Python con el framework de tu selección.
Subes el proyecto a un repositorio git. Y nos compartes el enlace.
Documentas tu proyecto con un README.

Nos indicas el tiempo involucrado para resolver el problema.

1. Solve the next problem prioritizing *performance*.

The median of ***M*** numbers is defined as the middle number after sorting them in order if ***M*** is odd. Or it is the average of the middle two numbers if ***M*** is even. You start with an empty number list. Then, you can add numbers to the list, or remove existing numbers from it. After each add or remove operation, output the median.

Example:

For a set of ***M* = 5** numbers **9, 2, 8, 4, 1** the median is the third number in the sorted set **1, 2, 4, 8, 9**, which is **4**. Similarly, for a set of ***M* = 4** numbers, **5, 2, 10, 4**, the median is the average of the second and the third element in the sorted set **2, 4, 5, 10**, which is **(4 + 5) / 2 = 4.5.**

Input:

The first line is an integer, ***N***, that indicates the number of operations. Each of the next ***N*** lines is either a x or r x. a x indicates that ***x*** is added to the set, and r x indicates that ***x*** is removed from the set.

Output:

For each operation: If the operation is add, output the median after adding ***x*** in a single line. If the operation is remove and the number ***x*** is not in the list, output Wrong! in a single line. If the operation is remove and the number ***x*** is in the list, output the median after deleting ***x*** in a single line. (If the result is an integer DO NOT output decimal point. And if the result is a real number, DO NOT output trailing 0s.)

Note: If your median is 3.0, print only 3. And if your median is 3.50, print only 3.5. Whenever you need to print the median and the list is empty, print Wrong!

Constraints:

$0 < N \leq 10^5$

For each a x or r x, ***x*** will always be a signed integer (which will fit in 32 bits).

Sample Input:

```
7
r 1
a 1
a 2
a 1
r 1
r 2
r 1
```

Sample Output:

```
Wrong!
1
1.5
1
1.5
1
Wrong!
```

Note: As evident from the last line of the input, if after remove operation the list becomes empty, you have to print *Wrong!*.

2. Solve next problem.

You are given an unrooted tree of ***n*** nodes numbered from **1** to ***n***. Each node ***i*** has a color, c_i.

Let ***d(i, j)*** be the number of different colors in the path between node ***i*** and node ***j***. For each node ***i***, calculate the value of sum_i, defined as follows:

$$sum_i = \sum_{j=1}^{n} d(i,\ j)$$

Your task is to print the value of sum_i for each node $1 \le i \le n$.

Input Format

The first line contains a single integer, ***n***, denoting the number of nodes.

The second line contains ***n*** space-separated integers, $c_1, c_2, \ldots. c_n$, where each c_i describes the color of node ***i***.

Each of the ***n* - 1** subsequent lines contains **2** space-separated integers, ***a*** and ***b***, defining an undirected edge between nodes ***a*** and ***b***.

Constraints

$1 \leq n \leq 10^5$
$1 \leq c_i \leq 10^5$

Output Format

Print ***n*** lines, where the $\boldsymbol{i}^{th}$ line contains a single integer denoting $\boldsymbol{sum_i}$.

Sample Input

5

1 2 3 2 3

1 2

2 3

2 4

1 5

Sample Output

10

9

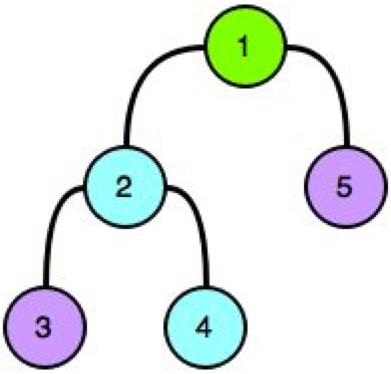
11

9

12

Explanation

The Sample Input defines the following tree:



Each $\boldsymbol{sum_i}$ is calculated as follows:

1. $sum_1 = d(1, 1) + d(1, 2) + d(1, 3) + d(1, 4) + d(1, 5) = 1 + 2 + 3 + 2 + 2 = 10$
2. $sum_2 = d(2, 1) + d(2, 2) + d(2, 3) + d(2, 4) + d(2, 5) = 2 + 1 + 2 + 1 + 3 = 9$
3. $sum_3 = d(3, 1) + d(3, 2) + d(3, 3) + d(3, 4) + d(3, 5) = 3 + 2 + 1 + 2 + 3 = 11$
4. $sum_4 = d(4, 1) + d(4, 2) + d(4, 3) + d(4, 4) + d(4, 5) = 2 + 1 + 2 + 1 + 3 = 9$
5. $sum_5 = d(5, 1) + d(5, 2) + d(5, 3) + d(5, 4) + d(5, 5) = 2 + 3 + 3 + 3 + 1 = 12$

Preguntas adicionales:

- Cuales serian las cualidades para un código limpio?
- Cuales serian los estándares según tú para un buen proyecto?
- Qué patrones conoce? y utiliza?