Purple Sheep And The Golden Apple Rush

Lior Yehezkely (lior5654)

1 Problem Statement

Momo, Rushing for Sarah's Hidden Golden Apple Treasure

Appleshire is a mythical kingdom, ruled by the almighty queen *Sarah Bitan*. In recent years, *Appleshire* has been through a *Golden Apple Rush*, due to a surprising increase in the growth rate of golden apples throughout the kingdom.

Appleshire consists of N towns, labeled 1 to N, connected by bi-directional roads. It is possible to travel from each town to every other town using the roads. Suprisingly, *Appleshire* has exactly $N-1$ such roads. Note that between each pair of towns there is at most one road.

Some of the towns of *Appleshire* are *abandoned*. *Abandoned* towns are towns connected to exactly one other town to save resources. All non-*abandoned* towns are connected to more than one town to improve the residents' life quality.

After the first few days of the *Golden Apple Rush*, *Sarah Bitan* has acquired many golden apples. Unfortunately, *Sarah* had not enough place to store them in her castle! Cleverly, *Sarah* decided to utilize the *abandoned* towns to store the precious apples.

In each *abandoned* town labeled i, *Sarah* decided to hide $L_i \geq 1$ golden apples in a treasure chest. A few years have passed since then, and the chests remained hidden.

Using *Appleshire*'s roads comes at a cost. each non-*abandoned* city sells *travel passes*, the cost of which is 1 golden apple. When buying a *travel pass*, if you already posses one, you must throw it away. Holding a travel pass is required for road usage. For each non-*abandoned* town labeled i, let $L_i \geq 1$ denote the *value* of the travel passes sold at i. The cost of using a road is v golden apples, where v is the value of the travel pass the user has. The user may use the travel pass as many times as they wish to, even in roads not connected directly to the city the travel pass was bought at. **Please note the difference in the definition of L_i for *abandoned* and non-*abandoned* towns.**

Momo is a purple-skinned sheep that has recently moved to *Appleshire*. *Momo* has just found out about *Sarah*'s treasures, and decided to go collect exactly one of them.

Let $u \in [1, N]$. *Momo* starts his journey at the town labeled u. *Momo* must only use *Appleshire*'s roads to travel between towns. Furthermore, *Momo* may buy multiple travel passes throughout the journey. Once *Momo* reaches an *abandoned* town he must collect the treasure hidden by *Sarah Bitan* at that town. *Momo* may not collect other treasures. ***Momo* may use the same road more than once during his journey!**

The *spending* of a journey is the total number of golden apples *Momo* would have to pay during the journey (both when buying new travel passes and when using a road), minus the number of apples *Momo* recieves at the end of the journey (the value of the treasure located at the *abandoned* city *Momo* reached). Note that **the *spending* may be negative, 0 or positive**.

Momo loves competitive programming, and decided to convert his story to a challenge. *Momo*'s challenge is the following: Given the road structure of *Appleshire* and the L values of all towns, determine **for each possible starting city u, the minimal possible *spending* of a valid journey**.

2 Input

The first line contains a single integer N ($2 \leq N \leq 4000$), the number of towns in *Appleshire*. The next line contains N space-separated integers $L_1, \ldots, L_N$ ($\forall i, 1 \leq L_i \leq 10^9$) as defined in problem statement. The next $N-1$ lines denote *Appleshire*'s roads. Each of these lines contains two space-separated integers u, v ($1 \leq u \neq v \leq N$), meaning there is a road between the towns labeled u and v.

3 Output

Output consists of a single line containing N space-separated integers. The i-th integer ($1 \leq i \leq n$) should be the minimal *spending* given $u = i$.

4 Samples

Sample Input 1	Sample Output 1

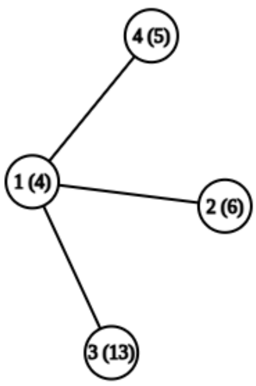
`4`
`4 6 13 5`
`1 2`
`1 3`
`1 4`
	`-8 -6 -13 -5`

Sample Input 2	Sample Output 2

`5`
`50 10 100 100 100`
`1 2`
`2 3`
`3 4`
`4 5`
	`-50 -69 32 1 -100`

5 Explanation

5.1 Sample 1



Appleshire 's Structure - Sample 1

In this sample, there are $N = 4$ towns.

In the above figure, the nodes denote *Appleshire* 's towns and the edges denote *Appleshire* 's roads.

For each town node, the number inside the parentheses indicates the L value of that town, and the number outside the parentheses indicates the town label.

For $u = 1$, *Momo* must do the following:

- Buy a travel pass (cost $= 1$). The value of the bought travel pass is 4.
- Move to one of the 3 neighbors, all of which are *abandoned* (cost $= 4$)
- Collect the hidden treasure at the chosen town.

In order to minimize the *spending*, *Momo* should select the *abandoned* town with the largest number of hidden apples - town 3. Hence, the minimal *spending* is $1 + 4 - L_3 = 5 - 13 = -8$.

For $u = 2, 3, 4$, since u is *abandoned*, *Momo* only has 1 option - collecting the hidden treasure at u. Hence, in these cases the minimal *spending* is $-L_u$.