Efficient managing

[Due to SPOJ restrictions, this problem has been modified with respect to the original version used in the Argentinian Programming Tournament of 2016 in order to have multiple test cases per input file. The original version of this problem (in Spanish) can be found at http://torneoprogramacion.com.ar/wp-content/uploads/2016/09/tap2016.pdf]

Nlogononia's train network consists of **N** stations, each of them strategically placed in a different city of the kingdom. Certain pairs of stations are connected by railways, and on each railway there is a train service going on both directions. For centuries the *Institute for the Cities' Perfect Connection* (ICPC) has been in charge of optimizing public transportation in Nlogonia, so today the train network is so efficient there is exactly one way to travel between any pair of cities using trains. Note that it may be necessary for a traveller to successively take several trains, in case there is no direct connection between the pair of stations he is traveling to/from. In other realms this might be considered an inconvenience, but Nlogonia's inhabitants are happy knowing that they will not waste a single minute thinking about which route to take from one city to another.

The ticket for each train service has certain price, so that a passenger taking one or more trains to travel between two cities has to buy the corresponding tickets before boarding each train. Nlogonia's currency is also extremely efficient, as there are bills with values corresponding to every non-negative power of two. This is, the denomination of bills in Nlogonia is of $2^0 = 1$ unit, $2^1 = 2$ units, $2^2 = 4$ units, and so on. As a result of this monetary eficiency, Nlogonians always pay their tickets by providing the minimum number of bills with which it is possible to reach the exact amount of the ticket they are buying.

To speed up ticket buying, the *Agency for Counting Money* (ACM) would like to introduce the following offer. When a passenger is going to make a trip, he can pay all the tickets he will be needing in advance. When doing so, he must present all the bills he would use along his journey, and the ACM will only take one bill of each denomination for which the passenger provided an odd amount of bills. In this way, if for example a passenger wants to buy three tickets with prices of **3**, **7** and **10** units, he will present two bills for the first ticket (with denominations **1** and **2**), three bills for the second one (with denominations **1**, **2** and **4**) and two bills for the third (with denominations **2** and **8**). The ACM would then take only one of the bills with denomination **2**, along with each of the bills with denominations **4** and **8**, returning to the passenger two bills of denomination **1**, as well as two bills with denomination **2**.

Now the ACM's steering committee is worried because this offer might be too expensive for the kingdom's treasury. There is certainly good cause for this, as one should note that it is even possible to travel for free (*e.g.* any round trip will be free, as an even number of tickets of each value will always be required). Your job is to evaluate the extent of the problem, for which the ACM has commanded you to determine the maximum price a passenger might have to pay when starting his journey from each of the **N** train stations in Nlogonia.

Input

There are multiple test cases in the input file. The first line contains an integer number **N**,

representing the number of train stations in Nlogonia ($2 \leq N \leq 10^5$). Train stations in Nlogonia are numbered from **1** to **N**. Each of the following **N-1** lines contains three integer numbers **A**, **B** and **C**, indicating there is a railway directly connecting stations **A** and **B**, being **C** the price of the ticket for the train service operating on said railway ($1 \leq A, B \leq N$ and $1 \leq C \leq 10^9$, with **A ≠ B**). The description of the railway network is always such that for each pair of distinct stations there exists one and only one sequence of train services connecting them.

Output

For each test case, print **N** lines each containing one integer number. The number printed on the **i**-th line should correspond to the maximum value of the tickets that may be payed by a passenger starting his journey at the station identified by the number **i**, when the offer described in the problem statement is applied.

Example

Input #1:
```
4
1 2 3
2 3 7
3 4 10
```

Output #1:
```
14
13
10
14
```

Input #2:
```
6
1 2 1
3 2 2
2 4 3
4 5 4
4 6 5
```

Output #2:
```
7
7
5
5
7
7
```

Input #3:
```
7
1 2 1
```

1 3 2
1 4 3
1 5 4
1 6 5
1 7 6

Output #3:
6
7
7
7
7
7
7