

Problem C. Beads and wires

Input file: `beads.in`
 Output file: `beads.out`
 Time limit: 1 second
 Memory limit: 128 megabytes

A popular child game at the time of Leonardo was called Beads-and-wires. Not surprisingly, it was played with beads and wires. The wires are red or blue. The beads are numbered from 1 to n . The game starts with a single bead. From this moment on, new beads can be added using wires as follows.

- $\text{Append}(w, v)$: a new bead w is attached to an existing bead v through a piece of red wire.
- $\text{Insert}(w, u, v)$: a new bead w is inserted by substituting an already existing red wire connecting existing beads u and v , with two new blue wires by putting w between u and v ; in other words, the existing red wire $u - v$ is replaced by two new blue wires $u - w$ and $w - v$.

Every piece of wire (both blue or red) has a certain length. At the end of the game, what counts is the sum of lengths of blue wires only (the length of red wires doesn't count): this is called the final score.

You are given the final configuration reached by a beads-and-wire game, specifying how beads are connected to one another and providing also the length of each piece of wire, but omitting the color of the wires.

You have to write a program that finds the maximal possible final score for that configuration. More precisely, among all the feasible beads-and-wire games that end with that configuration, you have to find one that has the maximum final score (sum of blue wire length) and you have to output that value.

Input

First line contains positive integer n — number of beads, which are numbered from 1 to n . Next $n - 1$ lines contains 3 numbers each: a_i, b_i ($1 \leq a_i < b_i \leq n$) and $1 \leq c_i \leq 10000$ — two beads connected by a piece of wire, and the length of that piece.

Output

Output one integer — the maximum final score.

Examples

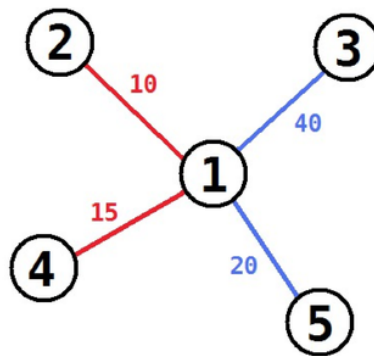
<code>beads.in</code>	<code>beads.out</code>
5 1 2 10 1 3 40 1 4 15 1 5 20	60
10 4 10 2 1 2 21 1 3 13 6 7 1 7 9 5 2 4 3 2 5 8 1 6 55 6 8 34	140

Note

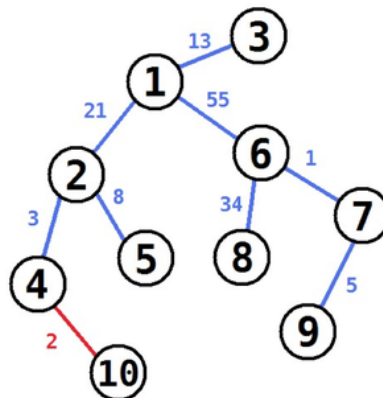
For the first sample testcase, the final score of 60 can be obtained as follows: start with the single bead 3.

- append 5 to 3 (with a wire of arbitrary length).
- insert 1 on the wire 3 – 5 (using wires of length 40 and 20).
- append 2 to 1 with a wire of length 10.
- append 4 to 1 with a wire of length 15.

The configuration obtained this way is shown in the first figure. It is possible to see that there is no way to obtain the same configuration (apart for the colors) with a larger final score.



For second sample testcase the configuration obtained this way is shown in the second figure, and has a final score of 140.



Scoring

Your program will be tested on 4 sets of input instances as follows:

Subtask 1 (points: 13)

$1 \leq n \leq 10$.

Subtask 2 (points: 15)

$1 \leq n \leq 200$.

Subtask 3 (points: 29)

$1 \leq n \leq 10000$.

Subtask 4 (points: 43)

$1 \leq n \leq 200000$.