Ropes Escape (ropes)

Edoardo was hacking a mainframe when he was traced by the cyberpolice (he didn't disconnect quickly enough). He is now in jail... but already planning his escape!

In fact, Edoardo managed to get a hold on *The Whip*, an evil tool that his captors regularly use to punish prisoners. The rope is made by a handle and a set of N interconnected ropes which form a tree: at the root of this tree we find the whip's handle (numbered 0), and each rope $i = 1 \dots N$ is directly tied to another rope P_i called *parent rope* (or to the handle if $P_i = 0$).

Edoardo's plan is to *untie* some of the ropes (one by one) from their parent rope and then *tie them back* to some other rope on the whip in order to form a single, very long rope (the handle should remain on one end of the rope). This rope can then be used to get out of the jail, as many prisoners did with bed sheets over the course of history!

Figure 1: What *The Whip* looks like.

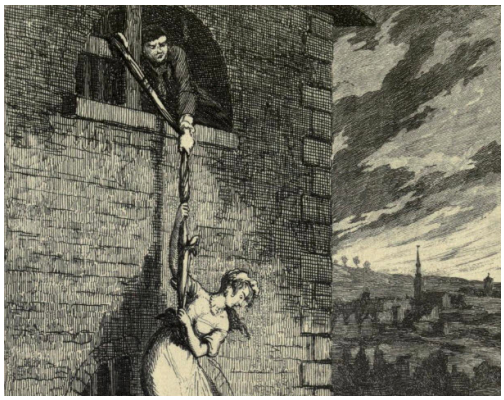


Figure 2: Prisoners escaping using bed sheets.

It's important to note that Edoardo can disconnect a rope from its parent even if there are other ropes connected to it or to the parent: he will then obtain a "sub-whip" that can then be reattached anywhere in the original whip.

Unfortunately, the ropes are tied together very strongly (after all those years of frantic usage) and untying rope i from its parent requires T_i seconds. Tying back a rope requires no time in comparison. Help Edoardo plan his escape by computing the minimum total time needed to transform *The Whip* into an escape rope!

☞ Among the attachments of this task you may find a template file `ropes.*` with a sample incomplete implementation.

Input

The first line contains the only integer N. The second line contains the N integers P_i. The third line contains the N integers T_i.

Output

You need to write a single line with an integer: the total time required for turning the whip into a straight rope that Edoardo can use to escape.

Constraints

- $1 \leq N \leq 100\,000$.
- $0 \leq P_i \leq N$ for each $i = 1 \ldots N$.
- $0 \leq T_i \leq 1000$ for each $i = 1 \ldots N$.
- All ropes are ultimately tied to the handle (*The Whip* is a single connected piece).

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) Examples.

– **Subtask 2** (10 points) All ropes are directly tied to the whip's handle: $P_i = 0$ for all i.

– **Subtask 3** (20 points) $N = 3$.

– **Subtask 4** (30 points) Untying times are fixed: $T_i = 1$ for all i.

– **Subtask 5** (20 points) $N \leq 10$.

– **Subtask 6** (20 points) No additional limitations.

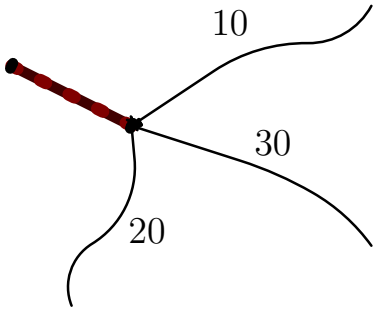
Examples

input	output
3 0 0 0 10 30 20	30
3 0 1 0 10 10 100	10

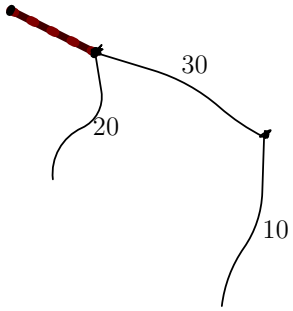
Explanation

In the **first sample case**, all ropes are directly connected to the handle. The best strategy is to:

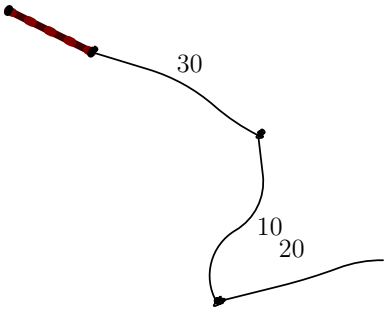
- disconnect rope 1 from the handle,
- connect rope 1 to rope 2,
- disconnect rope 3 from the handle,
- connect rope 3 to rope 2.



Initial situation.

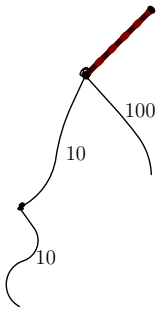


Untie rope 1 and connect to 2: 10 seconds.

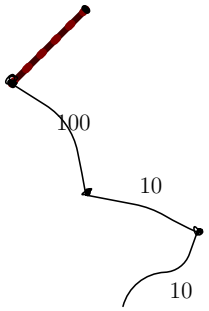


Untie rope 3 and connect to 1: 20 seconds.

In the **second sample case** we can make a straight rope by disconnecting rope 1 from the handle (obtaining the sub-whip of ropes 1 and 2), and then reconnect it to rope 3, in a total of 10 seconds.



Initial situation.



Untie rope 1 and connect to 2: 10 seconds.