2011 AIR

Fortunes have been won and lost within the cutthroat world of pigeon racing. Since it is not possible to prevent people looking up into the sky and observing the birds training — besides which spotting has a long tradition within the community and *spotties* (as they like to be called) are a determined bunch — owners resort to other methods to disguise their bird's form.

Courses consist of waypoints and birds are required to navigate between specified start and end waypoints. Along the way a bird must fly from waypoint to waypoint, only travelling between permitted pairings; each journey taking a fixed amount of time (based on the bird's ability and prevalent conditions). Birds are allowed to fly a permitted pairing as many times as they desire and their route only finishes when they stop flying — ideally by landing.

In competition a bird would calculate and fly the fastest overall route; thus testing a bird's veracity, mental capacity and dexterity. Some birds when training, to throw off the spotties, exercise their duplicity by flying the *second-fastest* route; this *must* take longer than the fastest route.

For example, suppose it takes a bird 3s to fly 1 → 2, 1s to fly 1 → 3, 4s to fly 2 → 4 and 7s to fly 3 → 4. The fastest route from 1 to 4 is 1 → 2 → 4 in a total of 7s, and the second-fastest is 1 → 3 → 4 in 8s. If it was also possible to fly directly 1 → 4 in 7s, giving tied routes with the fastest time, the second-fastest would still be the 8s route.

Write a program that finds the second-fastest route between two waypoints. The first line of the input will be a single integer w ($2 \leq w \leq 256$) indicating the number of waypoints. Each successive line will consist of three integers, the first two (between 1 and w inclusive) indicating a permitted pairing of two distinct waypoints, followed by the length of time it takes to traverse that route (between 1 and 1,000,000 inclusive). No pairing of waypoints will be duplicated and pairings can be travelled in either direction. The input will be terminated by the line **`-1 -1 -1`**.

You should output a single integer, the total amount of time required to traverse the second-fastest route between waypoints 1 and w. You will always be given test data where such a route exists.

SAMPLE INPUT

```
4
1 2 3
2 4 4
1 3 1
3 4 7
-1 -1 -1
```

SAMPLE OUTPUT

```
8
```