
F008A. Badenic dates

P68836_en

In the planet Badenia is used a system of dates different to ours. Although a date is identified like in our case with a day, a month and a year, we do not know how many months has a year, nor how many days has a month. However, it should not be an impediment to solve this problem:

Given a sequence of badenic dates, you must write the first date that it is between its previous one and its posterior one in the sequence. If there are not any, indicate it.

For instance, 32/15/2007 is between 23/2/1981 and 1/1/10234, because is greater than the first date and less than the second date. Another instance, 10/10/2 is between 100/10/2 and 100/1/2, because is greater than the first date and less than the second date.

Using the definition

```
struct Date {  
    int day, month, year;  
};
```

your program must include and use the function

```
bool less_than (const Date& d1, const Date& d2);
```

that returns true if and only if the date d1 is strictly less than date d2.

Input

The input starts with a natural number $n \geq 3$. Followed by n different badenic dates, each one described with three natural numbers separated by a slash, in this order: day, month and year

Output

Your program must print the first date that it is between its previous one and its posterior one in the sequence. If there is not any, it must indicate it. Follow the format of the instance.

Observation

You can not use vector to solve this problem.

Sample input 1

```
4 1/1/2008 23/2/1981 32/15/2007 1/1/10234
```

Sample output 1

```
32/15/2007
```

Sample input 2

```
3 100/10/2 10/10/2 100/1/2
```

Sample output 2

```
10/10/2
```

Sample input 3

```
3 1/1/1 3/3/3 2/2/2
```

Sample output 3

```
no date found
```

Problem information

Author : Professorat de P1

Translator : Carlos Molina

Generation : 2024-05-02 22:03:24

© *Jutge.org*, 2006–2024.

<https://jutge.org>