3303 - Songs

Europe - Southeastern - 2005/2006

John Doe is a famous DJ and, therefore, has the problem of optimizing the placement of songs on his tapes. For a given tape and for each song on that tape John knows the length of the song and the frequency of playing that song. His problem is to record the songs on the tape in an order that minimizes the expected access time. If the songs are recorded in the order $S_{(s1)}, \ldots, S_{s(n)}$ on the tape then the function that must be minimized is

$$\sum_{i=1}^{n} f_{s(i)} \sum_{j=1}^{s(i)} l_{s(j)}$$

where $f_{s(i)}$ is the frequency of playing the i -th song and l is the length of the song. Can you help John?

Input

The program input is from a text file. Each data set in the file stands for a particular set of songs that must be recorded on a tape. A data set starts with the number N (fits a 16 bit integer) of songs. Follow N the song specifications, and in the end, a number representing the position of a song S on the optimized tape. A song specification consists of the song identifier (fits an integer), the length of the song (fits a 16 bit integer), and the frequency of playing the song (a floating-point number). The program prints the identifier of the song S .

White spaces can occur freely in the input. The input data are correct and terminate with an end of file.

Output

For each set of data the program prints the result to the standard output from the beginning of a line.

Note: An input/output sample is in the table below. There is a single data set that contains 5 song specifications. The first song has the identifier 1, length 10 and playing frequency 45.5 etc. The result for the data set is the identifier of the 3rd song on the optimized tape. It is 2 for the given example.

Sample Input

```
5
1      10     45.5
2      5      20
30     20     10
400    50     35
15     17     89.9
3
```

Sample Output

```
2
```

Southeastern 2005-2006