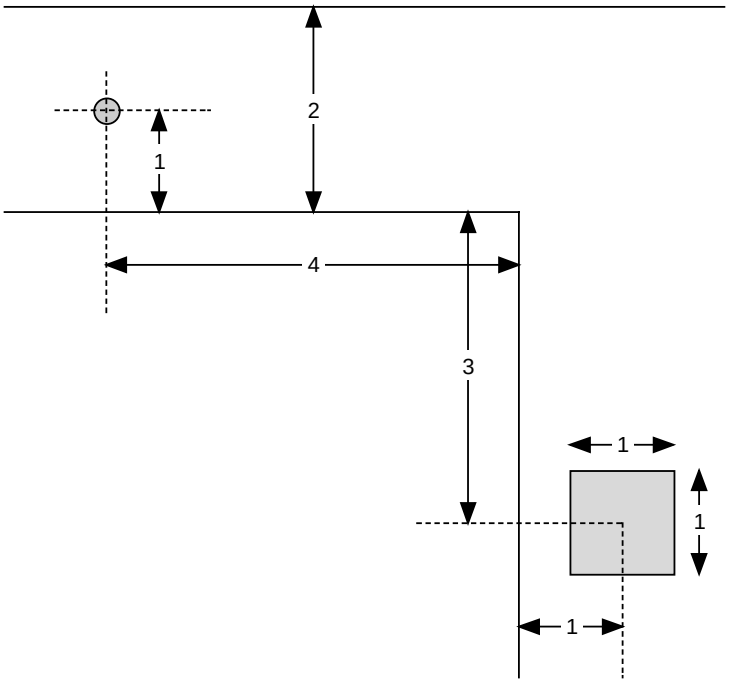
The ACM International Collegiate Programming Contest 1998–1999
North-western Europe Regional Contest
Sponsored by IBM

Problem C

Kroket

It is well-known throughout the world that the Dutch wear wooden shoes. Less well-known is the fact that one of their favourite games is played with this peculiar footwear: the ancient game of Kroket. Kroket is a mixture of football and golf. The objective is to kick a wooden ball (this is where the wooden shoes come in handy) with a diameter of 10–20 cm from a one square meter kick-off patch into a hole in the ground. But to make things more interesting, the playing ground is L-shaped and lined by walls 50 cm high. The ball will only fall into the hole if the centre of the ball passes exactly over the centre of the hole.

See the diagram below for a layout of the playing field, together with the dimensions (in metres) of the playing field.



Note that there is no wall to the right of the kick-off patch, so it is not wise to kick ball that way. There is also no wall left of the hole. The only way to kick the ball into the hole is to let it bounce against the top and bottom walls a couple of times. But how many times will it bounce?

This is where you come in again. Write a program to calculate the minimum number of bounces required to get the ball into the hole, and the distance the ball travels in that case. You may assume that

the ball is perfectly round, rolls perfectly straight, and bounces perfectly off both walls with angles of entry and exit being exactly equal (known as Snell's Law, named after Snellius, a famous Dutchman!).

Input

The input consists of a line containing the number of configurations N ($1 \leq N \leq 10000$). It is followed by N lines, one for each configuration. This line consists of:

- The diameter of the ball in centimetres: a real number ranging from 10 to 20;
- The kick-off position of the centre of the ball relative to the lower left corner of the kick-off area, in metres: two real co-ordinates (horizontal and vertical) ranging from 0 to 1.

Output

The output consists of one line per configuration containing the minimum number of bounces (an integer number), one space, and the distance the ball travels (a real number in decimal notation with three decimal digits).

Sample input

```
15
10.0 1.0 0.5
10.0 0.9 0.5
10.0 0.8 0.5
10.0 0.7 0.5
10.0 0.6 0.5
10.0 0.5 0.5
12.0 0.5 0.5
14.0 0.5 0.5
16.0 0.5 0.5
18.0 0.5 0.5
20.0 0.5 0.5
20.0 0.4 0.5
20.0 0.3 0.5
20.0 0.2 0.5
20.0 0.0 0.5
```

Output for the sample input

```
4 12.838
5 14.540
5 14.503
6 16.254
6 16.223
7 18.008
7 17.874
7 17.739
7 17.605
7 17.471
8 19.067
9 20.786
10 22.518
11 24.260
17 34.891
```