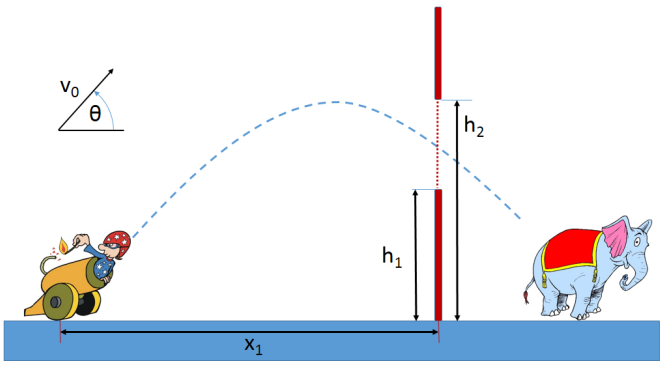
The Amazing Human Cannonball

Problem ID: humancann
CPU Time limit: 1 secon
Memory limit: 1024 MB
Difficulty: 1.4

Author: Godmar Back
Source: 2015 Virginia Tec School Programming Cor

The amazing human cannonball show is coming to town, and you are asked to double-check their calculations to make sure no one gets injured! The human cannonball is fired from a cannon that is a distance x_1 from a vertical wall with a hole through which the cannonball must fly. The lower edge of the hole is at height h_1 and the upper edge is at height h_2. The initial velocity of the cannonball is given as v_0 and you also know the angle θ of the cannon relative to the ground.



Source: picgifs.com

Thanks to their innovative suits, human cannonballs can fly without air resistance, and thus their trajectory can be modeled using the following formulas:

$$x(t) = v_0 t \cos\theta$$
$$y(t) = v_0 t \sin\theta - \frac{1}{2} g t^2$$

where $x(t), y(t)$ provides the position of a cannon ball at time t that is fired from point $(0, 0)$. g is the acceleration due to gravity ($g = 9.81$ m/s^2).

Write a program to determine if the human cannonball can make it safely through the hole in the wall. To pass safely, there has to be a vertical safety margin of 1 m both below and above the point where the ball's trajectory crosses the centerline of the wall.

Input

The input will consist of up to 100 test cases. The first line contains an integer N, denoting the number of test cases that follow. Each test case has 5 parameters: $v_0\ \theta\ x_1\ h_1\ h_2$, separated by spaces. v_0 $(0 < v_0 \leq 200)$ represents the ball's initial velocity in m/s. θ is an angle given in degrees $(0 < \theta < 90)$, x_1 $(0 < x_1 < 1000)$ is the distance from the cannon to the wall, h_1 and h_2 $(0 < h_1 < h_2 < 1000)$ are the heights of the lower and upper edges of the wall. All numbers are floating point numbers.

Output

If the cannon ball can safely make it through the wall, output "`Safe`". Otherwise, output "`Not Safe`"!

Sample Input 1

```
11
19 45 20 9 12
20 45 20 9 12
25 45 20 9 12
20 43 20 9 12
20 47.5 20 9 12
20 45 17 9 12
20 45 24 9 12
20 45 20 10 12
20 45 20 9 11
20 45 20 9.0 11.5
20 45 18.1 9 12
```

Sample Output 1

```
Not Safe
Safe
Not Safe
Not Safe
Not Safe
Not Safe
Not Safe
Not Safe
Not Safe
Safe
Safe
```