

## 11422 Escaping from Fractal Bacterium

A fractal bacterium is growing very fast and you have to calculate whether you are safe from its attack or not.

The growth of our bacteria is like a very well known fractal “Koch Snowflake” or “Koch Island”. The growth of it is shown below:

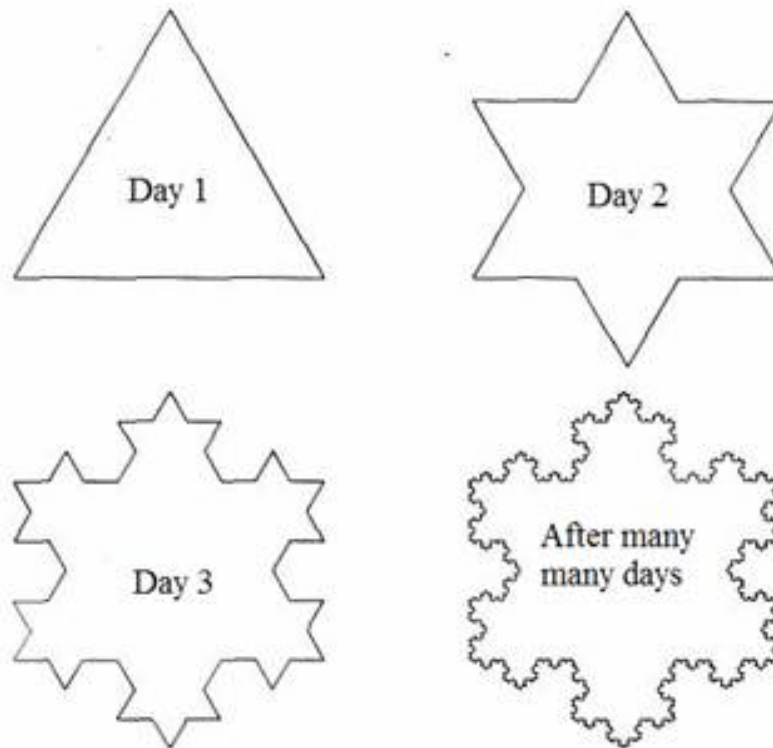


Figure: Growth of fractal bacteria

As you can see that at day one the bacteria has the shape of an equilateral triangle. In day two the all sides are extended outwards with the shape of another equilateral triangle whose sides are  $1/3$  rd of the triangle of day 1. This process is repeated again in day 3 and onwards.

The scientists have started working on a medicine and they are confident about finding means of killing the bacteria within 2 weeks. So if this bacterium cannot reach you within 2 weeks (Day 14) then you are safe. Given your location and the initial position of the bacterium, you will have to find whether you are safe or not.

### Input

First line of the input file contains an integer  $N$  ( $0 < N < 6001$ ) which denotes the number of input sets. Each of the next  $N$  lines makes the input for one set. Each line contains eight floating-point numbers  $x_1, y_1, x_2, y_2, x_3, y_3, x$  and  $y$  ( $|x_1|, |y_1|, |x_2|, |y_2|, |x_3|, |y_3|, |x|, |y| \leq 400000$ ). Here  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  are the coordinates of three vertexes of an equilateral triangle. This equilateral triangle denotes the position of the virus on the very first day and  $(x, y)$  is your location in Cartesian coordinate system. From the very next day (Day 2) this bacteria will start to grow following the rule mentioned in the problem statement. All the floating point number in the input file has 10 digits after the decimal

point. The inputs will be such that precision error will not be a problem if you use double-precision floating point numbers.

## Output

For each line of input produce one line of output. If the bacteria can reach you within Day 14 then you should produce a line ‘The bacterium destroys you within day  $d$ . ESCAPE!!’, where  $d$  is the number of days the bacteria takes to reach you. If the bacteria cannot reach you within day 14 (inclusive) then produce the line ‘You are safe :)’ instead (If you are still confused — your program should never print a number greater than 14 but can print 14 sometimes :).

**Note for the Sample:** Although it appears from the sample input that input for each set is scattered in two lines, actually those two lines is a single line as shown below:

```
3
45233.0000000000 149056.0000000000 146297.0000000000 75838.0000000000 159173.6480142890 199970.9914080705 68004.3137563194 159230.3729762797
43726.0000000000 134821.0000000000 196858.0000000000 195926.0000000000 67373.5177017519 297989.7021323186 33994.5456040129 164685.1050602284
149787.0000000000 67376.0000000000 39536.0000000000 17919.0000000000 137492.5183949670 -52832.6667926381 58599.6651155748 35914.0994623880
```

## Sample Input

```
3
45233.0000000000 149056.0000000000 146297.0000000000 75838.0000000000
    159173.6480142890 199970.9914080705 68004.3137563194 159230.3729762797
43726.0000000000 134821.0000000000 196858.0000000000 195926.0000000000
    67373.5177017519 297989.7021323186 33994.5456040129 164685.1050602284
149787.0000000000 67376.0000000000 39536.0000000000 17919.0000000000
    137492.5183949670 -52832.6667926381 58599.6651155748 35914.0994623880
```

## Sample Output

```
The bacterium destroys you within day 1. ESCAPE!!
The bacterium destroys you within day 3. ESCAPE!!
The bacterium destroys you within day 5. ESCAPE!!
```