

Code Jam 2010 - Round 1C

Load Testing

Problem

Now that you have won Code Jam and been hired by Google as a software engineer, you have been assigned to work on their wildly popular programming contest website.

Google is expecting a lot of participants (**P**) in Code Jam next year, and they want to make sure that the site can support that many people at the same time. During Code Jam 2010 you learned that the site could support at least **L** people at a time without any errors, but you also know that the site can't yet support **P** people.

To determine how many more machines you'll need, you want to know within a factor of **C** how many people the site can support. This means that there is an integer **a** such that you know the site can support **a** people, but you know the site can't support **a * C** people.

You can run a series of *load tests*, each of which will determine whether the site can support at least **X** people for some integer value of **X** that you choose. If you pick an optimal strategy, choosing what tests to run based on the results of previous tests, how many load tests do you need in the worst case?

Input

The first line of the input gives the number of test cases, **T**. **T** lines follow, each of which contains space-separated integers **L**, **P** and **C** in that order.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1) and y is the number of load tests you need to run in the worst case before knowing within a factor of **C** how many people the site can support.

Limits

Time limit: 30 seconds per test set.

Memory limit: 1GB.

$1 \leq T \leq 1000$.

$2 \leq C \leq 10$.

L, **P** and **C** are all integers.

Small dataset (Test set 1 - Visible)

$1 \leq L < P \leq 10^3$.

Large dataset (Test set 2 - Hidden)

$1 \leq L < P \leq 10^9$.

Sample

Sample Input

```
4
50 700 2
19 57 3
1 1000 2
24 97 2
```

Sample Output

```
Case #1: 2
Case #2: 0
Case #3: 4
Case #4: 2
```

Explanation

In Case #2, we already know that the site can support between 19 and 57 people. Since those are a factor of 3 apart, we don't need to do any testing.

In Case #4, we can test 48; but if the site can support 48 people, we need more testing, because $48 * 2 < 97$. We could test 49; but if the site can't support 49 people, we need more testing, because $24 * 2 < 49$. So we need two tests.