

De-RNG-ed

Problem

I want to make an online poker website. A very important component of such a system is the random number generator. It needs to be fast and random enough. Here is a compromise I came up with. I need a way to generate random numbers of length at most **D**. My plan is to select a prime number $P \leq 10^D$. I am also going to pick non-negative integers **A** and **B**. Finally, I'm going to pick an integer seed **S** between 0 and $P-1$, inclusive.

To output my sequence of pseudo-random numbers, I'm going to first output **S** and then compute the new value of **S** like this:

$$S := (A * S + B) \bmod P.$$

Then I will output the new value of **S** as the next number in the sequence and update **S** again by using the same formula. I can repeat this as many times as I want.

Do you think that this is a good random number generator? Can you write a program that takes **K** consecutive elements of a sequence that was generated by my random number generator, and prints the next element of the sequence?

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each one starts with a line containing **D** and **K**. The next line contains **K** consecutive elements generated by a random number generator of the kind described above.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1) and y is either the next number in the sequence, or the string "I don't know." if the answer is ambiguous.

Limits

Time limit: 30 seconds per test set.

Memory limit: 1GB.

$$1 \leq T \leq 100.$$

$$1 \leq K \leq 10.$$

The **K** integers will be consecutive elements of a sequence generated by a random number generator of the type described above.

Small dataset (Test set 1 - Visible)

$$1 \leq D \leq 4.$$

Large dataset (Test set 2 - Hidden)

$$1 \leq D \leq 6.$$

Sample

Sample Input

```
3
2 10
0 1 2 3 4 5 6 7 8 9
3 1
13
1 5
6 6 6 6 6
```

Sample Output

```
Case #1: 10
Case #2: I don't know.
Case #3: 6
```