

Brute-force Algorithm EXTREME

Please click [here](#) to download a PDF version of the contest problems. The problem is problem B in the PDF. But the data limits is slightly modified: $1 \leq P \leq 1000000$ in the original description, but in this EXTREME version, $1 \leq P \leq 1000000000$.

=====(EDIT, Francky)=====

Professor Brute is not good at algorithm design. Once he was asked to solve a path finding problem. He worked on it for several days and finally came up with the following algorithm:

```
Function Find(integer n,function func)
  If n=1
    For i = 1 to a do func()
  Elseif n=2
    For i = 1 to b do func()
  Else Find(n-1,Find(n-2,func))
Function Main
  Find(n,funny)
```

Any fool but Brute knows that the function “funny” will be called too many times. Brute wants to investigate the number of times the function will be called, but he is too lazy to do it.

Now your task is to calculate how many times the function “funny” will be called, for the given a, b and n. Because the answer may be too large, you should output the answer module by P.

Input

There are multiple test cases. The first line of the input contains an integer T, meaning the number of the test cases.

For each test cases, there are four integers a, b, P and n in a single line. You can assume that $1 \leq n \leq 1000000000$, $1 \leq P \leq 1000000$, $0 \leq a, b < 1000000$.

Output

For each test case, output the answer with case number in a single line.

Example

Input:

```
3
3 4 10 3
4 5 13 5
3 2 19 100
```

Output:

```
Case #1: 2
Case #2: 11
Case #3: 12
```