



Task 3: CATS

Esoteric languages are programming languages built around weird concepts either to test the limits of programming or for the purpose of comedy. One fine day, Mr. Panda was playing around with an esoteric language aptly called Counter and Two Stacks (CATS), whose name describes its basic usage. The language is designed to manipulate a counter (“COUNTER”) and two stacks (“S1” and “S2”) using several basic operations.

Under CATS, both stacks are initialised with an infinite number of 0’s. Data can be pushed and popped from the stacks using the operations “PUSH” and “POP” respectively. An additional operation, “ADD”, removes the top 2 elements of a stack, adds them and pushes the result back into the stack. Mr. Panda makes a novice attempt to draft some code that when given a query of 3 integers X , L and N , should print the X -th multiple of N larger than L . He uses negative numbers in his code and for convenience, Mr. Panda does not consider cases where L is a multiple of N .

Being new to programming, Mr. Panda’s code does not print the correct answer. Furthermore, he did not know that the stacks in this language were not designed to work with negative integers. In CATS, whenever the code attempts to push a negative number using the PUSH operation, the number is not pushed into the stack. Instead CATS reacts unconventionally and all the numbers in that stack will have their last binary digit flipped (note that the stack contains an infinite number of values). For example, 4, whose binary representation is 100, would become 5, whose binary representation is 101, and vice versa. Flipping the last binary digit of a number X is equivalent to taking $(X \text{ xor } 1)$ in Pascal or $(X \hat{=} 1)$ in C/C++.

In addition, Mr. Panda’s code mistakenly runs some operations twice. Eventually, Mr. Panda’s code can be described by the pseudocode below. Here, “T1” and “T2” represent the top elements of S1 and S2 respectively.



```
COUNTER = X
WHILE COUNTER > 0
    S2 PUSH T1          //Push the top element of S1 onto S2
    S1 POP              //Pop the top element of S1
    FLIP LAST BINARY BIT OF ALL NUMBERS IN S1
    IF T2 > L
        COUNTER = COUNTER - 1
        IF COUNTER == 0 PRINT T2
    ELSE
        S2 PUSH N
        S2 PUSH N
        S2 ADD
        S2 ADD
        S1 PUSH T2
        S1 PUSH T2
        S2 POP
        S2 POP
```

Mr. Panda's code takes a long time to run and is thus difficult to debug. Can you help him figure out the output of his code quickly?

Input format

Your program must read from the standard input. Each input consists of multiple queries to Mr. Panda's code. The first line of each input contains a single positive integer Q which represents the number of queries. Each of the next Q lines contains 3 positive integers which represent the values of X , L and N respectively in a query. As stated above, L is not a multiple of N . An example is provided below:

```
2
4 5 2
18 6 4
```

Output format

For each query in the input, your program must write to the standard output the number printed by Mr. Panda's code. It is guaranteed that Mr. Panda's code prints exactly one integer for each query. Output the answer for different queries on different lines. For the above example, the output is:

```
8
9
```



Explanation

For the first query, the following lines trace the value of COUNTER and the values in S1 every time the WHILE loop is repeated. For simplicity, only the top 4 elements in S1 are shown. Here, the leftmost value on each line represents the top element of S1.

```
COUNTER: 4  
S1: 0000...
```

```
COUNTER: 4  
S1: 4411...
```

```
COUNTER: 4  
S1: 8850...
```

```
COUNTER: 3  
S1: 9411...
```

```
COUNTER: 2  
S1: 5000...
```

```
COUNTER: 2  
S1: 9911...
```

```
COUNTER: 1  
S1: 8000...
```

In the next run, COUNTER will reach 0 so 8 is printed.

Subtasks

The maximum execution time on an input instance is 1.5s. Your program will be tested on 6 sets of input instances. Recall that all Q , X , L , and N are positive integers, and thus greater or equal to 1. The input instances satisfy the following conditions:

1. (4 marks) $Q = 1$, $X \leq 1,000,000$, $L \leq 20$, $N \leq 20$.
2. (4 marks) $Q \leq 50$, $X \leq 1,000,000$, $L \leq 50$, $N \leq 50$.
3. (4 marks) $Q \leq 500$, $X \leq 1,000,000$, $L \leq 80$, $N \leq 80$.
4. (4 marks) $Q \leq 1,000$, $X \leq 1,000,000$, $L \leq 1,000,000$, $N \leq 1,000,000$.
5. (4 marks) $Q \leq 10,000$, $X \leq 2^{30} - 1$, $L \leq 2^{30} - 1$, $N \leq 2^{30} - 1$.
6. (5 marks) $Q \leq 100,000$, $X \leq 2^{60} - 1$, $L \leq 2^{60} - 1$, $N \leq 2^{60} - 1$.