Home · Info · Challenges · Past editions · Ranking

Challenge 11 - Toast

« Prev Next »

You have **N** piles of toast on your table with **M** slices of toast in each and a very sharp knife. For some reason, you can't stop cutting the toast. Every second, you cut one of the piles in half and put one of the halves on top of the other half or on top of the table, forming a new pile.

How many seconds do you need to have exactly **K** slices of toast on the table?

Input

The first line will contain an integer **C**, the number of cases for our problem.
C lines follow, each with three integers **N**, **M** and **K**.

Output

For each case, a line starting with "Case #x: " followed by the number of seconds necessary to reach the desired amount of slices of toast. In the event that it's impossible, write IMPOSSIBLE instead. Every line is followed by a new line character.

Examples

Case 1:	Case 2:	Case 3:	Case 4:	Case 5:	Case 6:	Case 7:
1 1 1	1 1 2	1 1 4	2 2 12	1 3 7	10 5 500	1234 13 13131313

In Case 1, you need 0 seconds.
In Case 2, you need 1 second: cut the toast, and put the half anywhere.
In Case 3, you need 2 seconds: following the first cut you have a pile with 2 slices, and following the second cut you have a pile with 4 slices.

In Case 4, you need 3 seconds. One of the solutions is: following the first cut the resulting piles are (2, 4); following the second cut (2, 8), and following the third cut (4, 8).
In Case 5, you can't reach exactly 7 slices.
In Case 6, you need 10 seconds.
In Case 7, you need 29 seconds.

Limits

- $1 \le$ **N, M, K** $\le 2^{63}$

Sample Input

```
7
1 1 1
1 1 2
1 1 4
2 2 12
1 3 7
10 5 500
1234 13 13131313
```

Sample Output

```
Case #1: 0
Case #2: 1
Case #3: 2
Case #4: 3
Case #5: IMPOSSIBLE
Case #6: 10
Case #7: 29
```

Test your code

You can test your program against both the input provided in the test phase and the input provided in the submit phase. A nice output will tell you if your program got the right solution or not. You can try as many times as you want to. Be careful with extra whitespaces, the output should be exactly as described.

Test your program against the input provided in the test phase

Download test input

Program output:

Seleccionar archivo Ningún archivo seleccionado

Submit test output

Test your program against the input provided in the submit phase

Download input

Program output:

Seleccionar archivo Ningún archivo seleccionado

Submit output

During the submit phase, in some problems, we might give your program harder inputs. As with the test token, a nice output will tell you if your program got the right solution or not. You can try as many times as you need.

In the actual contest you first need to solve the test phase before submitting the code, you must provide the source code used to solve the challenge and you can only submit once (once your solution is submitted you won't be able to amend it to fix issues or make it faster).

If you have any doubts, please check the info section.

« Prev Next »

Tweet about this! #TuentiChallenge6 Follow @TuentiEng