Task_5

Problem#1: solve It using function

FizzBuzz

According to Wikipedia, FizzBuzz is a group word game for children to teach them about division. This may or may not be true, but this question is generally used to ~~torture~~ screen young computer science graduates during programming interviews.

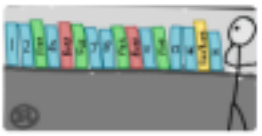


Image by chris morgan cc by

Basically, this is how it works: you print the integers from 1 to N, replacing any of them divisible by X with `Fizz` or, if they are divisible by Y, with `Buzz`. If the number is divisible by both X and Y, you print `FizzBuzz` instead.

Check the samples for further clarification.

Input

Input contains a single test case. Each test case contains three integers on a single line, X, Y and N ($1 \leq X < Y \leq N \leq 100$).

Output

Print integers from 1 to N in order, each on its own line, replacing the ones divisible by X with `Fizz`, the ones divisible by Y with `Buzz` and ones divisible by both X and Y with `FizzBuzz`.

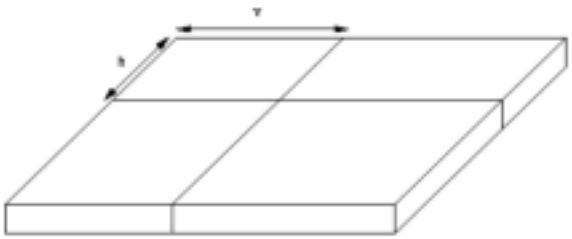
Problem#2:

Piece of Cake!

It is Greg's birthday! To celebrate, his friend Sam invites Greg and two other friends for a small party. Of course, every birthday party must have cake.

Sam ordered a square cake. She makes a single horizontal cut and a single vertical cut. In her excitement to eat cake, Sam forgot to make these cuts through the middle of the cake.

Of course, the biggest piece of cake should go to Greg since it is his birthday. Help Sam determine the volume of the biggest piece of cake that resulted from these two cuts.



Input

The input consists of a single line containing three integers n ($2 \leq n \leq 10\,000$), the length of the sides of the square cake in centimeters, h ($0 < h < n$), the distance of the horizontal cut from the top edge of the cake in centimeters, and v ($0 < v < n$), the distance of the vertical cut from the left edge of the cake in centimeters. This is illustrated in the figure above.

Each cake is 4 centimeters thick.

Sample Input 1	Sample Output 1
10 4 7	168

Sample Input 2	Sample Output 2
5 2 2	36

Problem#3:

Solve It using function

Apaxiaaaaaaaaaaaans!

The ancient and mysterious Apaxian civilization, which we most certainly did not make up, continues to confound the researchers at the Oriental Institute. It turns out that the Apaxians had a peculiar naming system: the more letters in your name, the higher your status in society. So, in Apaxian society, `robert` was probably a lowly servant, and `robertapalaxiamethostenes` was likely a High Priest or Minister. Even more than that, Apaxians valued the number of adjacent letters that were the same in a name. So, while `robert` continues to be an unimpressive name, `roooooooooobert` probably elicited cheers and applause wherever he went.

Unfortunately, this makes the task of reading Apaxian scrolls very cumbersome, especially when you consider that a particularly famous Apaxian queen had ten thousand consecutive `a`'s in her name. Legend has it that she was already two years old by the time the Royal Herald finished announcing her birth.

To make the Oriental Institute's life easier, the Department of Computer Science has offered to convert the Apaxian scrolls into a more readable format. Specifically, we will be taking Apaxian names and replacing all consecutive runs of the same letter by a single instance of such letter.

So, for example, the compact version of `roooobert` would be `robert`, where the four consecutive `o`'s have been replaced with a single `o`. Similarly, the compact version of `rrrooobbbert` would also be `robert`. On the other hand, the compact version of `robert` is still `robert`.

Input

The input contains a single name. Each name contains only lowercase letters (`a`–`z`), no whitespace, a minimum length of 1 character, and a maximum length of 250 characters.

Output

The output contains the compact version of the name: any time the same letter appears two or more times in sequence, it must be replaced by a single instance of that letter.

Sample Input 1

```
robert
```

Sample Output 1

```
robert
```

Sample Input 2

```
rooobert
```

Sample Output 2

```
robert
```