**Harry to Hogwarts**

Harry got a letter from Hogwarts School to join the wizardry. This letter consists of a magical key (6174) and a 4-digit number n with at least two distinct digits. Harry has to count in how many steps this number n can be converted to magical key invented by professor Dumbledore with this procedure as follows:

1.For number n, create two new numbers x and y consists of the digits in n in ascending and descending order respectively.

2.Subtract the smaller number from the larger number.

Help harry to count the steps to enter in the school.

For example:

N = 1234

Step 1: x = 4321 y= 1234 => n = x-y => 3087

Step 2: x = 8730 y= 0378 => n = x-y => 8352

Step 3: x = 8532 y= 2358 => n = x-y => 6174

and you are done.

Answer is 3

**Input Format**

integer four digit only >0

**Constraints**

None

**Output Format**

interger count

**Sample Input 0**

1235

**Sample Output 0**

7

**Sample Input 1**

1234

**Sample Output 1**

3

**Explanation 1**

* N = 1234
* Step 1: x = 4321 y= 1234 => n = x-y => 3087
* Step 2: x = 8730 y= 0378 => n = x-y => 8352
* Step 3: x = 8532 y= 2358 => n = x-y => 6174
* you are done.
* Answer is 3

## Color Combinations

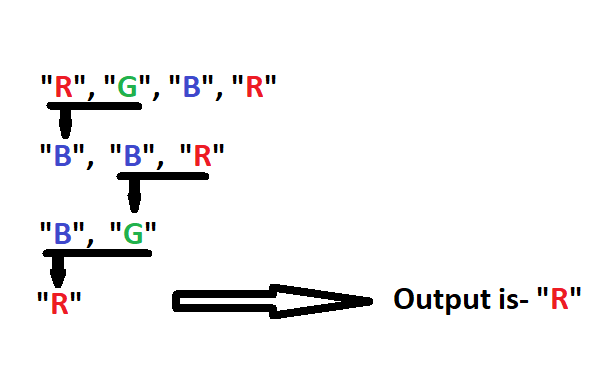
We have three colors: red, green, and blue in a list.

Replace the two different colors pair with third color in every dutye cycle of transformation.

Given N colors are in a line, determine the last remaining color after any possible sequence of such transformations.

For example, given the list is ['R', 'G', 'B', 'R']

and the following steps we need to take in transformation.



("R", "G") ---> "B"

("B", "G") ---> "R"

("B", "R") ---> "G"

**Input Format**

list of colors

**Constraints**

None

**Output Format**

Color

**Sample Input 0**

["R", "G", "B", "R", "G"]

**Sample Output 0**

"B"

**Explanation 0**

* ['R', 'G', 'B', 'R', 'G']
* {'G', 'R'}---> {'B'} ['B', 'B', 'R', 'G']
* {'R', 'B'}---> {'G'} ['B', 'G', 'G']
* {'G', 'B'}---> {'R'} ['R', 'G']
* {'G', 'R'}---> {'B'} ['B']
* Output representation : "B"

**Sample Input 1**

["R", "G", "B", "R", "G", "R", "B"]

**Sample Output 1**

"R"

**Explanation 1**

* ['R', 'G', 'B', 'R', 'G', 'R', 'B']
* {'G', 'R'}---> {'B'} ['B', 'B', 'R', 'G', 'R', 'B']
* {'R', 'B'}---> {'G'} ['B', 'G', 'G', 'R', 'B']
* {'G', 'B'}---> {'R'} ['R', 'G', 'R', 'B']
* {'G', 'R'}---> {'B'} ['B', 'R', 'B']
* {'R', 'B'}---> {'G'} ['G', 'B']
* {'G', 'B'}---> {'R'} ['R']