# A. Help Far Away Kingdom

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In a far away kingdom lived the King, the Prince, the Shoemaker, the Dressmaker and many other citizens. They lived happily until great trouble came into the Kingdom. The ACMers settled there.

Most damage those strange creatures inflicted upon the kingdom was that they loved high precision numbers. As a result, the Kingdom healers had already had three appointments with the merchants who were asked to sell, say, exactly $0.273549107$ beer barrels. To deal with the problem somehow, the King issued an order obliging rounding up all numbers to the closest integer to simplify calculations. Specifically, the order went like this:

- If a number's integer part does not end with digit $9$ and its fractional part is strictly less than $0.5$, then the rounded up number coincides with the number’s integer part.
- If a number's integer part does not end with digit $9$ and its fractional part is not less than $0.5$, the rounded up number is obtained if we add $1$ to the last digit of the number’s integer part.
- If the number’s integer part ends with digit $9$, to round up the numbers one should go to Vasilisa the Wise. In the whole Kingdom she is the only one who can perform the tricky operation of carrying into the next position.

Merchants found the algorithm very sophisticated and they asked you (the ACMers) to help them. Can you write a program that would perform the rounding according to the King’s order?

### Input

The first line contains a single number to round up — the integer part (a non-empty set of decimal digits that do not start with $0$ — with the exception of a case when the set consists of a single digit — in this case 0 can go first), then follows character «`.`» (a dot), and then follows the fractional part (any non-empty set of decimal digits). The number's length does not exceed $1000$ characters, including the dot. There are no other characters in the input data.

### Output

If the last number of the integer part is not equal to $9$, print the rounded-up number without leading zeroes. Otherwise, print the message "`GOTO Vasilisa.`" (without the quotes).

**Sample test(s)**

```
input
0.0
output
0
```

```
input
1.49
output
1
```

```
input
1.50
output
2
```

```
input
2.71828182845904523536
output
3
```

```
input
3.14159265358979323846
output
3
```

input

```
12345678901234567890.1
```

output

```
12345678901234567890
```

input

```
123456789123456789.999
```

output

```
GOTO Vasilisa.
```

# B. Help Chef Gerasim

time limit per test: 0.5 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

In a far away kingdom young pages help to set the table for the King. As they are terribly mischievous, one needs to keep an eye on the control whether they have set everything correctly. This time the royal chef Gerasim had the impression that the pages have played a prank again: they had poured the juice from one cup to another. Now Gerasim wants to check his hypothesis. The good thing is that chef Gerasim always pour the same number of milliliters of juice to all cups in the royal kitchen. Having thoroughly measured the juice in each cup, Gerasim asked you to write a program that will determine from which cup juice was poured to which one; otherwise, the program should determine that this time the pages set the table diligently.

To simplify your task we shall consider the cups to be bottomless so that the juice never overfills a cup and pours out, however much it can be. Besides, by some strange reason in a far away kingdom one can only pour to a cup or from one cup to another an integer number of milliliters of juice.

### Input

The first line contains integer $n$ — the number of cups on the royal table ($1 \leq n \leq 1000$). Next $n$ lines contain volumes of juice in each cup — non-negative integers, not exceeding $10^4$.

### Output

If the pages didn't pour the juice, print "`Exemplary pages.`" (without the quotes). If you can determine the volume of juice poured during exactly one juice pouring, print "$v$ `ml. from cup #`$a$ `to cup #`$b$`.`" (without the quotes), where $v$ represents the volume of poured juice, $a$ represents the number of the cup from which the juice was poured (the cups are numbered with consecutive positive integers starting from one in the order in which the cups are described in the input data), $b$ represents the number of the cup into which the juice was poured. Finally, if the given juice's volumes cannot be obtained using no more than one pouring (for example, the pages poured the juice from one cup to another more than once or the royal kitchen maids poured the juice into the cups incorrectly), print "`Unrecoverable configuration.`" (without the quotes).

**Sample test(s)**

input

```
5
270
250
250
230
250
```

output

```
20 ml. from cup #4 to cup #1.
```

input

```
5
250
250
250
250
250
```

output

```
Exemplary pages.
```

input

```
5
270
250
249
230
250
```

output

```
Unrecoverable configuration.
```

# C. Help Victoria the Wise

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Vasilisa the Wise from a far away kingdom got a present from her friend Helga the Wise from a farther away kingdom. The present is a surprise box, yet Vasilisa the Wise doesn't know yet what the surprise actually is because she cannot open the box. She hopes that you can help her in that.

The box's lock is constructed like that. The box itself is represented by an absolutely perfect black cube with the identical deepening on each face (those are some foreign nanotechnologies that the far away kingdom scientists haven't dreamt of). The box is accompanied by six gems whose form matches the deepenings in the box's faces. The box can only be opened after it is correctly decorated by the gems, that is, when each deepening contains exactly one gem. Two ways of decorating the box are considered the same if they can be obtained one from the other one by arbitrarily rotating the box (note that the box is represented by a perfect nanotechnological cube)

Now Vasilisa the Wise wants to know by the given set of colors the following: in how many ways would she decorate the box in the worst case to open it? To answer this question it is useful to know that two gems of one color are indistinguishable from each other. Help Vasilisa to solve this challenging problem.

### Input

The first line contains exactly $6$ characters without spaces from the set $\{\mathrm{R}, \mathrm{O}, \mathrm{Y}, \mathrm{G}, \mathrm{B}, \mathrm{V}\}$ — they are the colors of gems with which the box should be decorated.

### Output

Print the required number of different ways to decorate the box.

**Sample test(s)**

input

```
YYYYYY
```

output

```
1
```

input

```
BOOOOB
```

output

```
2
```

input

```
ROYGBV
```

output

```
30
```

# D. Help King

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

*This is the modification of the problem used during the official round. Unfortunately, author's solution of the original problem appeared wrong, so the problem was changed specially for the archive.*

Once upon a time in a far away kingdom lived the King. The King had a beautiful daughter, Victoria. They lived happily, but not happily ever after: one day a vicious dragon attacked the kingdom and stole Victoria. The King was full of grief, yet he gathered his noble knights and promised half of his kingdom and Victoria's hand in marriage to the one who will save the girl from the infernal beast.

Having travelled for some time, the knights found the dragon's lair and all of them rushed there to save Victoria. Each knight spat on the dragon once and, as the dragon had quite a fragile and frail heart, his heart broke and poor beast died. As for the noble knights, they got Victoria right to the King and started brawling as each one wanted the girl's hand in marriage.

The problem was that all the noble knights were equally noble and equally handsome, and Victoria didn't want to marry any of them anyway. Then the King (and he was a very wise man and didn't want to hurt anybody's feelings) decided to find out who will get his daughter randomly, i.e. tossing a coin. However, there turned out to be $n$ noble knights and the coin only has two sides. The good thing is that when a coin is tossed, the coin falls on each side with equal probability. The King got interested how to pick one noble knight using this coin so that all knights had equal probability of being chosen (the probability in that case should always be equal to $1 / n$). First the King wants to know the expected number of times he will need to toss a coin to determine the winner. Besides, while tossing the coin, the King should follow the optimal tossing strategy (i.e. the strategy that minimizes the expected number of tosses). Help the King in this challenging task.

### Input

The first line contains a single integer $n$ from the problem's statement ($1 \le n \le 10000$).

### Output

Print the sought expected number of tosses as an irreducible fraction in the following form: "$a / b$" (without the quotes) without leading zeroes.

**Sample test(s)**

input

```
2
```

output

```
1/1
```

input

```
3
```

output

```
8/3
```

input

```
4
```

output

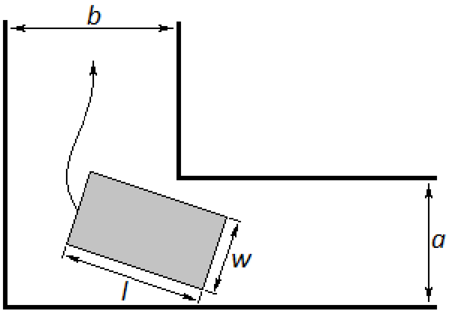
```
2/1
```

# E. Help Greg the Dwarf

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A very unusual citizen lives in a far away kingdom — Dwarf Gracula. However, his unusual name is not the weirdest thing (besides, everyone long ago got used to calling him simply Dwarf Greg). What is special about Dwarf Greg — he's been living for over 200 years; besides, he lives in a crypt on an abandoned cemetery and nobody has ever seen him out in daytime. Moreover, nobody has ever seen Greg buy himself any food. That's why nobody got particularly surprised when after the infernal dragon's tragic death cattle continued to disappear from fields. The people in the neighborhood were long sure that the harmless dragon was never responsible for disappearing cattle (considering that the dragon used to be sincere about his vegetarian views). But even that's not the worst part of the whole story.

The worst part is that merely several minutes ago Dwarf Greg in some unintelligible way got inside your house and asked you to help him solve a problem. The point is that a short time ago Greg decided to order a new coffin (knowing his peculiar character, you are not surprised at all). But the problem is: a very long in both directions L-shaped corridor leads to Greg's crypt, and you can't drag just any coffin through that corridor. That's why he asked you to help.



You've formalized the task on a plane like this: let the corridor's width before and after the turn be equal to $a$ and $b$ correspondingly (see the picture). The corridor turns directly at a right angle, the coffin is a rectangle whose length and width are equal to $l$ and $w$ ($l \geq w$) correspondingly. Dwarf Greg has already determined the coffin's length ($l$), which is based on his height; your task is to determine the coffin's maximally possible width ($w$), at which it can be brought to the crypt. Besides, due to its large mass (pure marble!) the coffin is equipped with rotating wheels; therefore it is impossible to lift it off the ground, however, arbitrary moves and rotations of the coffin in the plane become possible. The coffin may be rotated arbitrarily just before you drag it into crypt and move through the corridor.

Greg promised that if you help him, he will grant you immortality (I wonder how?). And if you don't, well... trust me, you don't want to know what happens if you don't help him...

### Input

The first line contains three space-separated integers $a$, $b$ and $l$ from the problem's statement ($1 \leq a, b, l \leq 10^4$).

### Output

Print the maximally possible width of a coffin with absolute or relative error no more than $10^{-7}$. If a coffin with the given length and positive width (the coffin that would meet the conditions from the problem's statement) does not exist, print "`My poor head =(`" (without quotes).

It is guaranteed that if the answer is positive, it will be not less than $10^{-7}$. All the hacks will also be checked to meet that condition.

**Sample test(s)**

input
```
2 2 1
```
output
```
1.0000000
```

input
```
2 2 2
```
output
```
2.0000000
```

input
```
2 2 3
```
output
```
1.3284271
```

input
```
2 2 6
```

output

```
My poor head =(
```

### Note

In the first example the answer is restricted by the coffin's length (remember — coffin's widths should not be larger than it's length).

In the second example it is possible to drag the coffin through the corridor thanks to rotating wheels: firstly, drag it forward by one side while it will not be hampered by the wall, then move it forward by adjacent side perpendicularly to the initial movement direction (remember — arbitrary moves and rotations of the coffin are possible).