Problem A. Archery

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

This is an interactive problem

Recently, a new variant of archery shooting competition was proposed — shooting with an unknown target.

The playfield consists of N targets arranged in a circle. The competition consists of Q rounds.

Before the start of the first round, the shooter can connect some pairs of targets with wires, while no two targets can be directly connected by more than one wire, and no target can be connected to itself. The connection remains unchanged during all rounds.

At the beginning of each round, one of the targets is secretly assigned as the main target, and the others are assigned as fake. The shooter does not know which target is selected as main. The shooter's task is to hit the main target in no more than 10 shots. If the shooter hits the main target, the scoreboard above the stand lights up green, and the round ends. If the shooter hits an fake target that is directly connected to the main target by a wire, the scoreboard lights up yellow. Otherwise, the scoreboard lights up red.

Your task is to write a program that connects the targets so that the target can be hit in no more than 10 shots in each of the Q rounds.

Interaction Protocol

The interaction starts with the jury program printing two integers N and Q — the number of targets and rounds, respectively ($1 \leq N, Q \leq 200$).

Then your program prints information about the targets connected by wires. First, print a line with a single integer K — the number of wires. Then print K lines, each containing two integers from 1 to N — the numbers of the targets connected by the corresponding wire. Two targets cannot be directly connected by more than one wire, and a target cannot be connected to itself.

After that, your program makes a shot, printing an integer from 1 to N — the number of the next target. The jury program responds with "`green`" if the shot hit the main target (after which you make the first shot of the next round or end the program if it was the last round), "`yellow`" if the shot hit a target directly connected to the main target, and "`red`" otherwise. After any of these two responses, your program continues shooting, choosing the next target. If more than 10 attempts are made in any round, the solution is considered incorrect.

Example

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td><pre>3 2

green

yellow

red

green</pre></td><td><pre>
2
1 2
1 3
1

1

2

3
</pre></td></tr>
</table>

Problem B. Backing Up The Password

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 512 megabytes

The Gnusmas company has proposed a new way to recall a pin code consisting of four digits. A randomly generated formula is displayed on the screen, in which the digits of the pin code are replaced with `a`, `b`, `c`, and `d`, as well as the value of this formula when the correct values of a, b, c, and d are substituted.

It is assumed that the user will recall the pin code after this. But in some cases, the formula may simply determine the answer unambiguously, which is unsafe - even someone who did not know the password will gain access. Or the formula will always be true - for example, a+b+c+d-a-b-c-d is always equal to zero, regardless of the pin code.

You are an intern at Gnusmas. To evaluate the quality of the hints, the system developers asked you to write a program that, given a hint, outputs the number of pin codes that satisfy it.

Input

The first line of the input contains a formula that the user entered to remember the pin code.

The formula may consist of:

- Variables `a`, `b`, `c`, and `d`, denoting the first, second, third, and fourth digits, respectively.
- Round brackets '`(`' and '`)`'.
- Integers from 1 to 10^4.
- Arithmetic operators '`+`', '`-`', '`*`'.

There are no spaces in the formula.

If several variables are written in a row without signs between them, this denotes a decimal number from the corresponding number of digits. For example, with $a = 0$, $b = 1$, and $c = 2$, $abcab$ denotes the number 01201 (or simply 1201, since leading zeros are ignored). This notation is called a *block*. It is guaranteed that the length of each block does not exceed 4. Otherwise, the formula complies with the rules of arithmetic (in particular, in the absence of brackets, multiplication takes precedence over addition and subtraction). Unary minus is absent.

A more formal description of the formula construction rule:

```
<number> :: an integer from 0 to 10000 without leading zeros
<var> :: a | b | c | d
<block> :: <var> | <var><var> | <var><var><var> | <var><var><var><var>
<op> :: + | - | *
<token> :: <block> | <number>
<formula> :: <token> | <sequence> <op> <sequence> | (<sequence>)
```

The formula is not empty, its length does not exceed 200. The formula contains no more than two multiplication signs.

The second line contains an integer N ($1 \le N \le 10^{16}$) - the value of the formula.

Output

Output the number of pin code variants for which the formula value S is equal to the number N. If there is only one variant, then output the pin code itself on the second line.

Examples

standard input	standard output
a+b+c+d 36	1 9999
a+b+c+d 5	56
b+c+d 27	10
abcd+abcd 1024	1 0512
1 2	0
abcd+1083 6006	1 4923

Problem C. Count The Repetitions

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 megabytes

Your task is to count the number of repetitions in the entered text. In the context of this task, a *repetition* is defined as a substring of a given string that can be represented as a concatenation of two identical strings. Equal substrings starting from different positions in the string are considered different. For example, in the text `hrhrhrhr` there is one repetition of length 8 (the entire string is a repetition: `hrhr`+`hrhr`), as well as 5 repetitions of length 4 (the substring `hrhr` at positions 1, 3, 5 and the substring `rhrh` at positions 2 and 4), for a total of 6 repetitions.

Count the number of repetitions in the given string.

Input

The input contains a single string s, consisting of lowercase Latin letters. The length of the string is at least 3 and at most 10^5 characters.

Output

Output a single integer — the number of repetitions in the given string.

Examples

standard input	standard output
`hrhrhrhr`	`6`
`hhhhhhh`	`12`
`voronezh`	`0`

Problem D. Difference in Games

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Participants in WTA tournaments play matches according to the following rules.

The game consists of games. Each game is won by one or another tennis player. As soon as one player has won at least 6 games and has won at least two more games than her opponent, it is considered that this tennis player has won a set, a new set starts and the game count starts again. The game is played until two sets are won.

The set score in a won match can be very different: 6:0 6:0, 6:4 0:6 14:12 and so on. You are given the difference in the number of games won in one match between the winner and the loser. Find out how many different matches there are under WTA rules with this difference. A match is considered different if at least one of the sets has a different score (e.g. 6:4, 6:3 and 6:3, 6:4 matches) are considered different — the scores in both sets are different). The winner of a match shall always be declared first.

Input

The input contains one integer — the difference d between the winner and the loser, in games ($-42 \le d \le 42$).

Output

If there is infinite number of games with the difference d, print "`oo`". Otherwise print one integer — the number of the distinct WTA games with the difference d.

Examples

standard input	standard output
4	oo
34	0

Note

In the Sample 1, any game $n : n-2$, $m : m-2$ for integer $m, n > 6$ have the difference 4, so the answer is infinite.

Problem E. Eligibility Test

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

In the Byteland Library was found the ancient book with the stories and legends about the semi-mythical wise queen Bytica and three her children. One of legends says that the Bytica's kingdom has a form of trapezoid of nonzero area with sides a, b, c and d. When the children grow, Bytica decided to split the kingdom to three triangular provinces to look at the children administrative skills as the governors and decide who is eligible to be the next queen or king.

To avoid the conflicts around size and form of the provinces, wise queen decided, that all parts shall be **equal** as the triangles, so two lines on the map was drawn and the children get their quests.

You want to check, if this legend can be true. Of course, you cant go to the archeological expedition during the contest, but you can check, if the situation is possible with the given values of a, b, c and d.

Input

Input contains four integers a, b, c and d, each integer on the new line ($1 \le a, b, c, d \le 1000$).

Output

If the trapezoid with the given length of the sides exists, have non-zero area and the queen can divide the kingdom to 3 equal triangles, print 1. Otherwise print 0.

Examples

standard input	standard output
6 2 3 4	1
1 2 4 8	0

Problem F. Fishing

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 megabytes

Stef, Klay, Draymond and their friends are fishing now.

They choose $N-1$ meter long river where N fish tanks are placed, one fish tank at each integer coordinate. Some tanks contain white fish, some tanks contain black fish, other tanks are empty.

The fishing is too boring, so Draymond found a funny trick. He can throw (from short distance, of course) the stone into river at any position (either at the some tank or between the tanks), and the wave goes at the same speed in both directions. When the wave reaches the tank with the fish, the fish in the tank jumps out the water for a moment and then falls back in the tank.

All fish have the same reaction time, that is, fish at the same distance from the place where the stone was thrown will jump simultaneously.

Stef and Klay especially likes when both a black and a white fish are jumping simultaneously and call it a combo.

Draymond wants to maximize the number of combos. He can choose the place for the stone and select the color of fish to be placed for the every empty tank (at the moment when stone will be thrown, all tanks shall contain fish). Draymond got W specimens of white fish and B specimens of the black fish to decide among them.

Input

The first line contains three integers N, W, and B ($2 \le N \le 50,000$, $0 \le W, B \le 50,000$), where N is the number of fish tanks, W and B are the number of extra white and black fish respectively.

The second line contains N characters each of which is one of 'W', 'B', or '?', representing how the fish tanks are originally populated. Here, 'W' means that the tank is occupied by the white fish, 'B' means it is black, and '?' means that you need to place a fish at that position.

The number of positions where Draymond needs to place a fish does not exceed $W + B$.

Output

Print one integer — the maximum number of combos the fishermen can see after Draymond populates all the tanks and throws the stone in the river.

Example

standard input	standard output
7 2 1 BBW?W??	3

Problem G. Game Analysis

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 512 megabytes

In the spin-off of the well-known movie Alice and Bob are playing the chess match. The match consists of several games and starts with the score 0:0. If the player wins the game, she is awarded with 1 point, otherwise (in case of a tie in game) players are awarded with 0.5 points each.

You are given the current score of the players. Find the total number of games already played and the minimal and maximal number of tied games.

Input

First line of the input contains one real number A — the score of Alice. The second line of the input contains one real number B — the score of Bob ($0 \le A, B \le 6$). The numbers are given without extra zeros after the decimal point (and without decimal point itself in case of the integer score). You may assume that the score A:B can be reached in the match after several games.

Output

In the first line print total number of games that are already played. In the second line — minimal possible number of the tied games. In the third line — maximal possible number of the tied games.

Examples

standard input	standard output
1.5 0.5	2 1 1
0 0	0 0 0

Problem H. Magic Squares

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Consider the integer matrix $n \times n$ as *magic square*, if it contains pairwise distinct positive integer and the sum of the integers in each horizontal, each vertical and each of two big diagonals is the same.

Consider the matrix $2k + 1 \times 2k + 1$ *hypermagic square* if:

- It contains all ingeters between 1 and $(2k + 1)^2$.
- It is the magic square.
- Any submartix built of integers that have the distance from the martix boundary t or less ($1 \le t \le k - 1$), is the magic square (you may represent those submatrices as the matrices with the same center and odd length of side).

Given the odd integer n, find any hypermagic square $n \times n$.

Input

Input contains one integer n ($5 \le n \le 999$, n is odd).

Output

Print n lines, each consisting n integers — the hypermagic square. If there are more than one solutions, print any of them.

Example

standard input	standard output
5	11 10 21 20 3 4 12 19 8 22 2 9 13 17 24 25 18 7 14 1 23 16 5 6 15

Problem I. Integer With All Digits

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Given integer N, find the maximal integer with the followng properties:

- The decimal representation of the integer have exactly N digits.
- The decimal representation contains each of digits between 0 and 9 at least once.
- The integer is divisible by 7.

Input

First line of the input contains one integer N ($1 \le N \le 10^9$).

Output

If there are no such integer, print -1. Otherwise print the answer — the integer with N digits in the **compact** form.

In the **compact** form, any $k \ge 5$ consecutive digits X are encoded as $(k)X$; for ex, 888889999 is encoded as (5)89999, $10^{100} - 1$ as (100)9, and 111122223333 or 232323232323 are encoded by self. You may assume that the answer length does not exceed 10^5 characters.

Examples

standard input	standard output
3	-1
10	9876543201
2023	(2014)9876542103

Problem J. Joyful Points

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Consider the point (a, b) with real coordinate *joyful*, if all values $a + b$, $a \cdot b$, $a - b$, $b - a$, a/b, b/a are defined and are **integer**.

Find the number of the joyful points (x, y) such that $l \leq a, b \leq r$.

Input

Input contains two integers l and r — the boundaries of the coordinates ($-10^9 \leq l \leq r \leq 10^9$).

Output

Print one integer — number of the joyful points in the given area. If there are infinity many joyful points, print -1 instead.

Example

standard input	standard output
0 1	1

Problem K. King of Cabbages

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

In the economics simulator game "King of Cabbages" the player buys factories with the goal of collecting as many cabbages as possible in the shortest time.

There are a total of N different types of factories in the game.

The number of cabbages f_0 increases every second first by C times, and then the number of cabbages produced by the factories of the type 1 (basic) is added. A factory of the first type produces K_1 cabbages every second, so if at time t you have f_0 cabbages and f_1 factories of the type 1, at time $t+1$ you will have $C \cdot f_0 + K_1 \cdot f_1$ cabbages.

The factories of type $i > 1$, however, are producing not the cabbages directly. They are producing factories!

The number of factories of type i is also increased every second first by C times, and then the number of factories of type i produced by factory of type $i+1$ is added. Factories of type $i+1$ (where i>0) every second produce $(i+1) \cdot K_{i+1}$ factories of type i, so if at time t you have f_i factories of type i and f_{i+1} factories of type $i+1$ at time $t+1$ you will have $C \cdot f_i + (i+1) \cdot K_{i+1} \cdot f_{i+1}$ factories of type i.

Given the amount of cabbages (f_0) and factories of each type at the time $t = 0$ and the values of C and K_i, determine the total number of cabbages at time $t = T$ modulo 998 244 353.

Input

The first line of the input contains three integers N ($1 \le N \le 2 \cdot 10^5$) — the number of the production types (i.e. cabbages and $N-1$ factory), the target time moment T ($1 \le T \le 10^9$) and the constant C ($1 \le C \le 10^9$).

The second line contains N integers — the initial values of $f_0, f_1, \ldots, f_{N-1}$, in order ($0 \le f_i \le 10^9$). The third line contains N integers $K_0, K_1, \ldots, K_{N-1}$, ($0 \le K_i \le 10^9$).

Output

Print one integer — the number of the cabbages after second T modulo 998 244 353.

Example

standard input	standard output
3 3 2 1 1 1 1 1 1	32

Problem L. Letters

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 1024 megabytes

Usually, the simplest task to learn the C language is the text output task "Hello World". Not surprisingly, the solution to this task can be found very easily on the Internet, which is a backdoor when you want to give the task to students on the fourth attempt to pass the exam...

As a result, the problem was modified — the students were required to write some other phrase of a length not exceeding 64. However, during the testing, some troubles appeared: instead of "Hello world" students could change the case of English letters arbitrarily, i.e. output "hEllo wOrLd", "HELLO WORlD" and so on. On the fourth attempt, such a solution is accepatable... but the teacher wants like to know the number of additional (not coinciding with the correct case) variants that can be accepted.

You are given the original phrase — a string consisting of spaces (code 32) and characters with codes from 33 to 122 inclusive. Find out how many **additional** answers you have to count as correct.

Input

Input contains one string L consisting of the characters with ASCII codes between 32 and 122, inclusively. You may assume that the first and the last character of the string are not spaces and that the length of the given string is positive and is not greater than 64.

Output

Print one integer — number of the answers that are not exactly equal with L, but differ only by the case of the English letters.

Examples

standard input	standard output
`Hello,world`	`1023`
`June 29, 2023`	`15`
`7!=5040`	`0`