Sexta Competencia

A. Fraction Formula

1.0 s, 256 MB

Mr. Potato Head has been promoted and now is a math professor at the UNAL.

For his first course he is willing to teach hard subjects, so at the moment he is teaching how to add and subtract fractions.

To complete his course the students have to do a long series of exercises, each exercise corresponds to a valid formula containing only additions and subtractions of fractions.

Formally a valid formula is one of the following:

- A fraction
- $F_1 + F_2$
- $F_1 - F_2$
- (F_1)

where F_1 and F_2 are also valid formulas.
Mr. Potato Head knows that the exam would be impossible if fractions are too large or if they are negative, so he decides that for every fraction a/b, $0 \leq a \leq 100$ and $0 < b \leq 20$.

Can you pass the course of Mr. Potato Head?

Input

The input consists of several lines, each line contains a valid formula without spaces.

It is guaranteed that all lines contains valid formulas and the total number of characters in all formulas does not exceed $2 * 10^5$

Output

For each formula output a line with an *irreducible* fraction $a/b, b > 0$ — The solution of the corresponding formula

input
```
1/2+1/3
1/5-2/10
1/2+(1/2-2/1)
```
output
```
5/6
0/1
-1/1
```

A fraction is *irreducible* if its numerator and denominator do not have common divisors greater than 1

B. Graduation

1.0 s, 256 MB

It is time for Mr. Potato Head, the best student in his high school, to enter the university, he has chosen to study in the UNAL. In order for Mr. Potato Head to finish his studies he must take and pass n courses.

In the UNAL every course can be prerequisite of *at most* another course, i.e. if a course A is a prerequisite of course B, Mr. Potato Head must pass the course A before taking the course B, when a course has no prerequisites it can be taken in any semester. Moreover, Mr. Potato Head knows that it is very hard to pass more than k courses in a semester, therefore, no matter how many courses he can take in a semester, he will take at most k.

Given the list of courses that Mr. Potato Head must pass and the information about prerequisites, which is the minimum number of semesters that Mr. Potato Head must take to graduate from UNAL?

Input

The first line of input consist of integers n $(1 \leq n \leq 10^4)$ and k $(1 \leq k \leq 10)$ — The number of courses Mr. Potato Head must pass and the maximum number of courses per semester Mr. Potato Head will take, respectively.

The following line consists of n integers $a_1, a_2, \ldots, a_n$ separated by spaces, where the i-th course is the prerequisite of the course a_i. If the i-th course is not a prerequisite of any other course $a_i = 0$.

Output

Print a single integer — The minimum number of semesters Mr. Potato Head needs to graduate.

input
```
4 2
3 3 4 0
```
output
```
3
```

input
```
3 3
0 1 2
```
output
```
3
```

It is guaranteed that Mr. Potato Head can graduate within the conditions above.

C. Kernel Of Love

10.0 s, 256 MB

Mr. Potato Head works on Unified Non-linear Algorithms about Love (UNAL). These algorithms are connected to a traditional machine learning branch called Kernel methods. Mr. Potato Head has discovered a Kernel function which measures the similarity of two persons and hence can predict the likelihood of them being a good couple. He has taken his discoveries one step forward, after running a Kernel algorithm over a vast database of Facebook profiles, he made some interesting (albeit scary) discoveries: every single human can be mapped bijectively to a Fibonacci number, which allowed him to derive a formula that tells if a couple will be happy for ever and ever.

The Fibonacci numbers are the sequence of numbers $\{F_k\}_{k=1}^{\infty}$ defined by the linear recurrence equation

$$F_{k+2} = F_{k+1} + F_k$$

with $F_1 = F_2 = 1$.

A perfect couple is represented by two numbers x and y such that:

1. x and y are Fibonacci numbers.
2. They are attractive to each other but not too much, this holds true when $gcd(x, y) = 1$
3. They are not too different or too similar, this is achieved when $(x + y) \mod 2 = 1$
4. Their eternal combination leads to another human being, this means, another Fibonacci number. This happens when $x + y = z$ where z is a Fibonacci number.

Mr. Potato Head is astonished with his discovery, he now wants to understand how many truly happy couples are there in the world. For a given n he wants to know how many couples exist on the first n human beings (i.e. the first n Fibonacci numbers) such that all conditions above hold true.

Input

The first line of the input represents the number of test cases. Each case consists of a single integer n $(1 \le n \le 10^5)$ per line.

Output

For each case print the number of perfect couples.

input

```
6
1
4
8
17
20
25
```

output

```
0
3
5
11
13
17
```

D. Integer Prefix

1.0 s, 256 MB

The Unique Numerical Association of Language (UNAL) is an association created to conserve only numbers in different texts. In fact, numbers are so important for UNAL, that its lemma is: 'people can communicate only using digits'.

Mr. Potato Head, the head of the UNAL has assigned you a task to help them to achieve the goal of the association.

Given a text T you have to find the longest non-empty prefix of T consisting only of digits, for the UNAL this is enough to understand the whole text.

Input

Input consists of a single line with a text T without spaces.

The length of the text is between 1 and $2 * 10^5$

Output

Print one line with the longest non-empty numeric prefix of T, if there is no such prefix print -1 instead.

input

```
23082019UNAL
```

output

```
23082019
```

input

```
_1234567890
```

output

```
-1
```

E. Boring Non-Palindrome

2.0 s, 256 MB

Holidays are almost over and our dear friend Mr. Potato Head wants to enjoy his last days of vacations with different fun activities like solving equations, calculating areas of figures, among others.

Currently, he is playing with different strings, although he finds that most of strings are boring non-palindromic strings. That is why Mr. Potato Head will transform any string into a palindrome inserting characters at the end of the corresponding string.

However, Mr. Potato Head does not want to spend his lasts days of holidays only inserting characters in strings, so he will insert the minimum possible of characters to convert the boring non-palindromic string into a fun palindromic one.

Can you guess the fun strings Mr. Potato Head created from boring ones?

Input

Input consists of a single line with a non-empty string without spaces.

The length of the string is at most 5000.

Output

Print one single line with the string after the changes of Mr. Potato Head.

input

```
helloworld
```

output

```
helloworldlrowolleh
```

input

```
anitalavalatina
```

output

```
anitalavalatina
```

A palindrome is a string that reads the same backwards as forwards

F. Cheap Kangaroo

1.0 s, 256 MB

There are N kangaroos going out to eat at an Indian restaurant. The i^{th} kangaroo wants to eat **exactly** x_i food. The kangaroos all want to order the same size of plates, but each one can order more than one plate for themselves if they need to. If the kangaroo orders more than he needs, he can simply hide the leftovers in his pouch.

At this Indian restaurant, the cost of the plate is the same as its size. Since Karl the Kangaroo is paying and is low on money, he wants to know what is the minimum cost to feed all N kangaroos and what is the largest possible size of the plates that satisfies this minimum cost?

Input

The first line of input is T – the number of test cases.

The first line of each test case is an integer N ($1 \le N \le 10^5$).

The second line contains N space-separated integers x_i ($1 \le x_i \le 10^9$).

Output

For each test case, output a line containing two space-separated integers – the minimum cost and the maximum plate size that corresponds to when the total cost is minimized.

input
```
2
1
5
2
4 2
```
output
```
5 5
6 2
```

G. Ayoub and Lost Array

1 second, 256 megabytes

Ayoub had an array a of integers of size n and this array had two interesting properties:

- All the integers in the array were between l and r (inclusive).
- The sum of all the elements was divisible by 3.

Unfortunately, Ayoub has lost his array, but he remembers the size of the array n and the numbers l and r, so he asked you to find the number of ways to restore the array.

Since the answer could be very large, print it modulo $10^9 + 7$ (i.e. the remainder when dividing by $10^9 + 7$). In case there are no satisfying arrays (Ayoub has a wrong memory), print 0.

Input

The first and only line contains three integers n, l and r ($1 \le n \le 2 \cdot 10^5, 1 \le l \le r \le 10^9$) — the size of the lost array and the range of numbers in the array.

Output

Print the remainder when dividing by $10^9 + 7$ the number of ways to restore the array.

input
```
2 1 3
```
output
```
3
```

input
```
3 2 2
```
output
```
1
```

input
```
9 9 99
```
output
```
711426616
```

In the first example, the possible arrays are : $[1, 2], [2, 1], [3, 3]$.

In the second example, the only possible array is $[2, 2, 2]$.

H. Colorful Bricks

2 seconds, 256 megabytes

On his free time, Chouti likes doing some housework. He has got one new task, paint some bricks in the yard.

There are n bricks lined in a row on the ground. Chouti has got m paint buckets of different colors at hand, so he painted each brick in one of those m colors.

Having finished painting all bricks, Chouti was satisfied. He stood back and decided to find something fun with these bricks. After some counting, he found there are k bricks with a color different from the color of the brick on its left (the first brick is not counted, for sure).

So as usual, he needs your help in counting how many ways could he paint the bricks. Two ways of painting bricks are different if there is at least one brick painted in different colors in these two ways. Because the answer might be quite big, you only need to output the number of ways modulo $998\,244\,353$.

Input

The first and only line contains three integers n, m and k ($1 \le n, m \le 2000, 0 \le k \le n - 1$) — the number of bricks, the number of colors, and the number of bricks, such that its color differs from the color of brick to the left of it.

Output

Print one integer — the number of ways to color bricks modulo $998\,244\,353$.

input
```
3 3 0
```
output
```
3
```

input
```
3 2 1
```
output
```
4
```

In the first example, since $k = 0$, the color of every brick should be the same, so there will be exactly $m = 3$ ways to color the bricks.

In the second example, suppose the two colors in the buckets are yellow and lime, the following image shows all 4 possible colorings.