A. Flipping Game

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

Iahub got bored, so he invented a game to be played on paper.

He writes n integers $a_1, a_2, \ldots, a_n$. Each of those integers can be either 0 or 1. He's allowed to do exactly one move: he chooses two indices i and j ($1 \le i \le j \le n$) and flips all values a_k for which their positions are in range $[i, j]$ (that is $i \le k \le j$). Flip the value of x means to apply operation $x = 1 - x$.

The goal of the game is that after **exactly** one move to obtain the maximum number of ones. Write a program to solve the little game of Iahub.

Input

The first line of the input contains an integer n ($1 \le n \le 100$). In the second line of the input there are n integers: $a_1, a_2, \ldots, a_n$. It is guaranteed that each of those n values is either 0 or 1.

Output

Print an integer — the maximal number of 1s that can be obtained after exactly one move.

Sample test(s)

input

```
5
1 0 0 1 0
```

output

```
4
```

input

```
4
1 0 0 1
```

output

```
4
```

Note

In the first case, flip the segment from 2 to 5 $(i = 2, j = 5)$. That flip changes the sequence, it becomes: [1 1 1 0 1]. So, it contains four ones. There is no way to make the whole sequence equal to [1 1 1 1 1].

In the second case, flipping only the second and the third element $(i = 2, j = 3)$ will turn all numbers into 1.

B. Prison Transfer

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

The prison of your city has n prisoners. As the prison can't accommodate all of them, the city mayor has decided to transfer c of the prisoners to a prison located in another city.

For this reason, he made the n prisoners to stand in a line, with a number written on their chests. The number is the severity of the crime he/she has committed. The greater the number, the more severe his/her crime was.

Then, the mayor told you to choose the c prisoners, who will be transferred to the other prison. He also imposed two conditions. They are,

- The chosen c prisoners has to form a contiguous segment of prisoners.
- Any of the chosen prisoner's crime level should not be greater then t. Because, that will make the prisoner a severe criminal and the mayor doesn't want to take the risk of his running away during the transfer.

Find the number of ways you can choose the c prisoners.

Input

The first line of input will contain three space separated integers n $(1 \le n \le 2 \cdot 10^5)$, t $(0 \le t \le 10^9)$ and c $(1 \le c \le n)$. The next line will contain n space separated integers, the i^{th} integer is the severity i^{th} prisoner's crime. The value of crime severities will be non-negative and will not exceed 10^9.

Output

Print a single integer — the number of ways you can choose the c prisoners.

Sample test(s)

input

```
4 3 3
2 3 1 1
```

output

```
2
```

input

```
1 1 1
2
```

output

```
0
```

input

```
11 4 2
2 2 0 7 3 2 2 4 9 1 4
```

output

```
6
```

C. Inna and Choose Options

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

There always is something to choose from! And now, instead of "Noughts and Crosses", Inna choose a very unusual upgrade of this game. The rules of the game are given below:

There is one person playing the game. Before the beginning of the game he puts 12 cards in a row on the table. Each card contains a character: "X" or "O". Then the player chooses two positive integers a and b $(a \cdot b = 12)$, after that he makes a table of size $a \times b$ from the cards he put on the table as follows: the first b cards form the first row of the table, the second b cards form the second row of the table and so on, the last b cards form the last (number a) row of the table. The player wins if some column of the table contain characters "X" on all cards. Otherwise, the player loses.

Inna has already put 12 cards on the table in a row. But unfortunately, she doesn't know what numbers a and b to choose. Help her win the game: print to her all the possible ways of numbers a, b that she can choose and win.

Input

The first line of the input contains integer t $(1 \le t \le 100)$. This value shows the number of sets of test data in the input. Next follows the description of each of the t tests on a separate line.

The description of each test is a string consisting of 12 characters, each character is either "X", or "O". The i-th character of the string shows the character that is written on the i-th card from the start.

Output

For each test, print the answer to the test on a single line. The first number in the line must represent the number of distinct ways to choose the pair a, b. Next, print on this line the pairs in the format axb. Print the pairs in the order of increasing first parameter (a). Separate the pairs in the line by whitespaces.

Sample test(s)

input

```
4
OXXXOXOOXOOX
OXOXOXOXOXOX
XXXXXXXXXXXX
OOOOOOOOOOOO
```

output

```
3 1x12 2x6 4x3
4 1x12 2x6 3x4 6x2
6 1x12 2x6 3x4 4x3 6x2 12x1
0
```

D. Palindromic Times

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

Tattah is asleep if and only if Tattah is attending a lecture. This is a well-known formula among Tattah's colleagues.

On a Wednesday afternoon, Tattah was attending Professor HH's lecture. At `12:21`, right before falling asleep, he was staring at the digital watch around Saher's wrist. He noticed that the digits on the clock were the same when read from both directions i.e. a palindrome.

In his sleep, he started dreaming about such rare moments of the day when the time displayed on a digital clock is a palindrome. As soon as he woke up, he felt destined to write a program that finds the next such moment.

However, he still hasn't mastered the skill of programming while sleeping, so your task is to help him.

Input

The first and only line of the input starts with a string with the format "`HH:MM`" where "`HH`" is from "`00`" to "`23`" and "`MM`" is from "`00`" to "`59`". Both "`HH`" and "`MM`" have exactly two digits.

Output

Print the palindromic time of day that comes soonest after the time given in the input. If the input time is palindromic, output the soonest palindromic time after the input time.

Sample test(s)

input
```
12:21
```
output
```
13:31
```

input
```
23:59
```
output
```
00:00
```

E. k-Tree

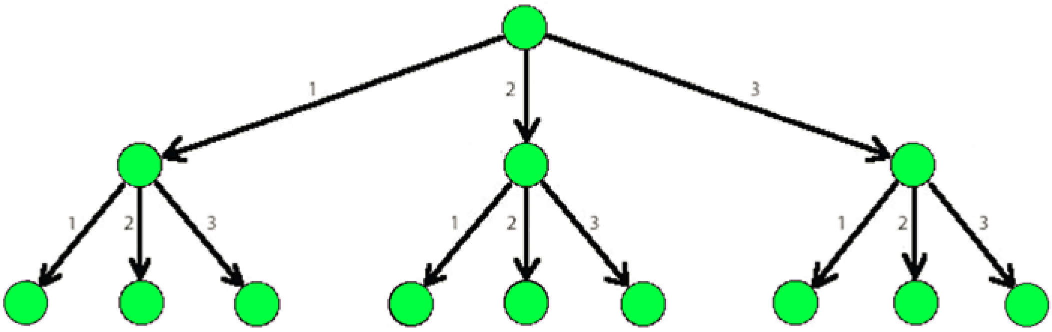
time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Quite recently a creative student Lesha had a lecture on trees. After the lecture Lesha was inspired and came up with the tree of his own which he called a k-tree.

A k-tree is an infinite rooted tree where:

- each vertex has exactly k children;
- each edge has some weight;
- if we look at the edges that goes from some vertex to its children (exactly k edges), then their weights will equal $1, 2, 3, \ldots, k$.

The picture below shows a part of a 3-tree.



As soon as Dima, a good friend of Lesha, found out about the tree, he immediately wondered: "How many paths of total weight n (the sum of all weights of the edges in the path) are there, starting from the root of a k-tree and also containing at least one edge of weight at least d?".
Help Dima find an answer to his question. As the number of ways can be rather large, print it modulo 1000000007 $(10^9 + 7)$.

Input

A single line contains three space-separated integers: n, k and d $(1 \le n, k \le 100; 1 \le d \le k)$.

Output

Print a single integer — the answer to the problem modulo 1000000007 $(10^9 + 7)$.

Sample test(s)

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

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

```
input
4 3 2
output
6
```

```
input
4 5 2
output
7
```

F. Queue

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

There are n walruses standing in a queue in an airport. They are numbered starting from the queue's tail: the 1-st walrus stands at the end of the queue and the n-th walrus stands at the beginning of the queue. The i-th walrus has the age equal to a_i.

The i-th walrus becomes displeased if there's a younger walrus standing in front of him, that is, if exists such j ($i < j$), that $a_i > a_j$. The *displeasure* of the i-th walrus is equal to the number of walruses between him and the furthest walrus ahead of him, which is younger than the i-th one. That is, the further that young walrus stands from him, the stronger the displeasure is.

The airport manager asked you to count for each of n walruses in the queue his displeasure.

Input

The first line contains an integer n ($2 \le n \le 10^5$) — the number of walruses in the queue. The second line contains integers a_i ($1 \le a_i \le 10^9$).

Note that some walruses can have the same age but for the displeasure to emerge the walrus that is closer to the head of the queue needs to be **strictly younger** than the other one.

Output

Print n numbers: if the i-th walrus is pleased with everything, print "-1" (without the quotes). Otherwise, print the i-th walrus's displeasure: the number of other walruses that stand between him and the furthest from him younger walrus.

Sample test(s)

input
```
6
10 8 5 3 50 45
```
output
```
2 1 0 -1 0 -1
```

input
```
7
10 4 6 3 2 8 15
```
output
```
4 2 1 0 -1 -1 -1
```

input
```
5
10 3 1 10 11
```
output
```
1 0 -1 -1 -1
```

G. Long Path

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

One day, little Vasya found himself in a maze consisting of $(n+1)$ rooms, numbered from 1 to $(n+1)$. Initially, Vasya is at the first room and to get out of the maze, he needs to get to the $(n+1)$-th one.

The maze is organized as follows. Each room of the maze has two one-way portals. Let's consider room number i $(1 \le i \le n)$, someone can use the first portal to move from it to room number $(i+1)$, also someone can use the second portal to move from it to room number p_i, where $1 \le p_i \le i$.

In order not to get lost, Vasya decided to act as follows.

- Each time Vasya enters some room, he paints a cross on its ceiling. Initially, Vasya paints a cross at the ceiling of room 1.
- Let's assume that Vasya is in room i and has already painted a cross on its ceiling. Then, if the ceiling now contains an odd number of crosses, Vasya uses the second portal (it leads to room p_i), otherwise Vasya uses the first portal.

Help Vasya determine the number of times he needs to use portals to get to room $(n+1)$ in the end.

Input

The first line contains integer n $(1 \le n \le 10^3)$ — the number of rooms. The second line contains n integers p_i $(1 \le p_i \le i)$. Each p_i denotes the number of the room, that someone can reach, if he will use the second portal in the i-th room.

Output

Print a single number — the number of portal moves the boy needs to go out of the maze. As the number can be rather large, print it modulo 1000000007 (10^9+7).

Sample test(s)

input

```
2
1 2
```

output

```
4
```

input

```
4
1 1 2 3
```

output

```
20
```

input

```
5
1 1 1 1 1
```

output

```
62
```

H. Match & Catch

time limit per test: 1 second
memory limit per test: 512 megabytes
input: standard input
output: standard output

Police headquarter is monitoring signal on different frequency levels. They have got two suspiciously encoded strings s_1 and s_2 from two different frequencies as signals. They are suspecting that these two strings are from two different criminals and they are planning to do some evil task.

Now they are trying to find a common substring of minimum length between these two strings. The substring must occur only once in the first string, and also it must occur only once in the second string.

Given two strings s_1 and s_2 consist of lowercase Latin letters, find the smallest (by length) common substring p of both s_1 and s_2, where p is a unique substring in s_1 and also in s_2. See notes for formal definition of substring and uniqueness.

Input

The first line of input contains s_1 and the second line contains s_2 $(1 \le |s_1|, |s_2| \le 5000)$. Both strings consist of lowercase Latin letters.

Output

Print the length of the smallest common unique substring of s_1 and s_2. If there are no common unique substrings of s_1 and s_2 print -1.

Sample test(s)

input
```
apple
pepperoni
```
output
```
2
```

input
```
lover
driver
```
output
```
1
```

input
```
bidhan
roy
```
output
```
-1
```

input
```
testsetses
teeptes
```
output
```
3
```

Note

Imagine we have string $a = a_1a_2a_3...a_{|a|}$, where $|a|$ is the length of string a, and a_i is the i^{th} letter of the string.

We will call string $a_la_{l+1}a_{l+2}...a_r$ $(1 \le l \le r \le |a|)$ the substring $[l, r]$ of the string a.

The substring $[l, r]$ is unique in a if and only if there is no pair l_1, r_1 such that $l_1 \ne l$ and the substring $[l_1, r_1]$ is equal to the substring $[l, r]$ in a.