CodeAvatar Test Series
Test- String

Date: ………………………………………………………… Name: …………………………………………………………………………

Marks Obtained: ……………………………………… Batch: …………………………………………………………………………

1. FIRST REVERSE

Using the function FirstReverse(**str**) take the **str** parameter being passed and return the string in reversed order. For example: if the input string is "Hello World and Coders" then your program should return the string **sredoC dna dlroW olleH**.

Sample Test Cases

Input:"coderbyte"

Output:"etybredoc"

Input:"I Love Code"

Output:"edoC evoL I"

Hint: Think of how you can loop through a string or array of characters backwards to produce a new string.

2. LONGEST WORD

Using the function LongestWord(**sen**) take the **sen** parameter being passed and return the largest word in the string. If there are two or more words that are the same length, return the first word from the string with that length. Ignore punctuation and assume **sen** will not be empty.

Sample Test Cases

Input:"fun&!! time"

Output:"time"

Input:"I love dogs"

Output:"love"

Hint: It might be easier to first convert the string into an array of words, and then loop through the array searching for the word with the most characters.

3. LETTER CHANGES

Using the function LetterChanges(**str**) take the **str** parameter being passed and modify it using the following algorithm. Replace every letter in the string with the letter following it in the alphabet (**ie.** c becomes d, z becomes a). Then capitalize every vowel in this new string (a, e, i, o, u) and finally return this modified string.

Sample Test Cases

Input:"hello*3"

Output:"Ifmmp*3"

Input:"fun times!"

Output:"gvO Ujnft!"

Hint: Changing a character to the next one that appears in the alphabet can easily be achieved by using the letters character code (ASCII).

4. LETTER CAPITALIZE

Using the function LetterCapitalize(**str**) take the **str** parameter being passed and capitalize the first letter of each word. Words will be separated by only one space.

Sample Test Cases

Input:"hello world"

Output:"Hello World"

Input:"i ran there"

Output:"I Ran There"

Hint: There might be a built-in function in your programming language that capitalizes the first letter of each word.

5. SIMPLE SYMBOL

Using the function SimpleSymbols(**str**) take the **str** parameter being passed and determine if it is an acceptable sequence by either returning the string **true** or **false**. The **str** parameter will be composed of **+** and **=** symbols with several letters between them (**ie.** ++d+===+c++==a) and for the string to be true each letter must be surrounded by a **+**symbol. So the string to the left would be false. The string will not be empty and will have at least one letter.

Sample Test Cases

Input:"+d+=3=+s+"

Output:"true"

Input:"f++d+"

Output:"false"

Hint: A simple loop through the string and checking if each letter is surrounded by a + symbol should work. You can also try writing a simple regular expression to solve this challenge.

6. TIME CONVERT

Using the function TimeConvert(**num**) take the **num** parameter being passed and return the number of **hours** and **minutes** the parameter converts to (**ie.** if **num** = 63 then the output should be 1:3). Separate the number of hours and minutes with a colon.

Sample Test Cases
Input:126
Output:"2:6"

Input:45
Output:"0:45"
Hint: Dividing the number by 60 will give you the number of hours, and then to get the minutes the modulo operation will be useful.

7. ALPHABET SOUP

Using the function AlphabetSoup(**str**) take the **str** string parameter being passed and return the string with the letters in alphabetical order (**ie.** hello becomes ehllo). Assume numbers and punctuation symbols will not be included in the string.

Sample Test Cases
Input:"coderbyte"
Output:"bcdeeorty"

Input:"hooplah"
Output:"ahhloop"
Hint: Using a built-in sort function on the string should do the trick for this challenge.

8. Compiler Version

You are converting an old code for a new version of the compiler.
In the old code we have used "->" for pointers. But now we have to replace each "->" with a ".". But this replacement shouldn't be done inside commments. A comment is a string that starts with "//" and terminates at the end of the line.
Input:
At max. 2000 lines of code.
Each line of code consists of at maximum 60 characters.
Output:
New code with required changes.

SAMPLE INPUT
```
int t; //variable t
t->a=0;  //t->a does something
return 0;
```
SAMPLE OUTPUT
```
int t; //variable t
t.a=0;  //t->a does something
return 0;
```

Explanation
"->" is not converted to "." inside comments.

9. Sumit's String

Given a string **'S'** , u need to tell whether it is **'sumit's string or not'**.
A string is called 'Sumit's String' , if distance between adjacent character is 1.
Consider that the alphabets are arranged in cyclic manner from **'a' to 'z'**. distance between any character **'x' and 'y'** will be defined as minimum number of steps it takes **'x' to reach 'y'**. Here, character **'x'** can start moving clockwise or anti-clockwise in order to reach at position where character **'y'** is placed.
Print **'YES'** if it is Sumit's string else print **'NO'**, for each yest case.

Input :

- test cases, t
- string , s

Output:
Desired O/p

Constraints :

- string length<=250
- string has only lower case letters

SAMPLE INPUT
```
3
aba
zza
bcd
```

SAMPLE OUTPUT
```
YES
NO
YES
```

Explanation
Sample Case 1: 'aba' is a Sumit's string , since all the adjacent characters are exactly 1 unit of distance apart.
Sample Case 2: 'zza' is not a Sumit's string since 'z' and 'z' are at the same position and hence 0 distance apart.

10. DNA Pride!

Everyone is familiar with Pratik's obsession with DNA and how much he likes to find the correct pair for the nucleotide bases. One day Samim found him exaggerating about his knowledge of DNA bases. So Samim challenged Pratik about finding the correct base pair for the given DNA sequence and show the result. Also **he secretly introduced some of RNA nucleotide bases** to test Pratik. Now initially he accepted the challenge but soon found out about how big the sequence actually was, so he came to you ask him for your in finding the sequence and keep his pride about the knowledge of DNA molecules.
You are given a string that contains the nucleotide bases of DNA and RNA, you are needed to find the correct pair for all the bases and print the corresponding sequence obtained. In case the sequence contains a RNA base, print **"Error RNA nucleobases found!"** (without quotes).

INPUT FORMAT
The first line of input contains **T**, the no of test cases. The next line of input contains **N**, the no of bases in each of the DNA sequence The line after that contains the DNA sequence.

OUTPUT FORMAT
For each test case output your answer on a new line.

CONSTRAIN
$1 \le T \le 10^4$
$1 \le N \le 10^6$

SAMPLE INPUT
```
3
2
AG
4
ATGC
6
UGCACT
```

SAMPLE OUTPUT
```
TC
TACG
Error RNA nucleobases found!
```

Explanation
For the second test case, we have the DNA sequence as **"ATGC"**. Now **the base adenine (A) pairs with thymine (T), thymine (T) pairs with adenine (A), guanine (G) pairs with cytosine (C), and cytosine (C) pairs up with guanine (G)**. Thus the DNA pair for the given sequence is **"TACG"**

11. VowelPhobia

Manish has got the task to frame a speech for his professor at the university at the Annual sports meet.But the problem is that the professor has speech dyslexia and he can't speak the words clearly which have vowels in them. So Manish has to avoid such words and has to minimise their usage in the speech letter. Your task is to help Manish mark the vowels in the words so that he can minimise their use. You are given a string S consisting of lower case letters only. You need to count the number of vowels in the string S.

INPUT The first line will contain an integer T denoting the number of test cases. The following T lines will contain a string S in lower case letters only.

OUTPUT Print the number the vowels in the string S.

CONSTRAINTS 1<=T<=100

SAMPLE INPUT
1
hashes

SAMPLE OUTPUT
2

12. Print first Occurence

Given a string containing only lower case letters ,print first occurrence of all the letters present in that order only.

Input :

- Test cases, t
- string ,s

Output :
Desired Output

Constraint :
string length <=200

SAMPLE INPUT
2
aasdvasvavda
sajhags

SAMPLE OUTPUT
asdv
sajhg

13. Conversion

Given a string, convert it into its number form .

- A or a -> 1
- B or b -> 2
- C or c -> 3

. . .

- Z or z -> 26
- space -> $

Input:

- test cases, t
- string , s

Output:
Desired O/p

Constraints: string length <=200

SAMPLE INPUT
2
AMbuj verma
Aaaa bBBB

SAMPLE OUTPUT
11322110$22518131

1111$2222

14. Pallindrome

You are given a string S.You need to check whether it's a palindrome or not. Print "YES" (without quotes) if S is a palindrome and "NO" (without quotes) if S is not a palindrome.
To know what a palindrome is you can check http://bit.ly/HashesPallindrome

INPUT First line will contain an integer T that is the number of test cases. The following T lines will contain: -A string S consisting of only lowercase letters.

OUTPUT Output YES or NO in new line for each test case.

SAMPLE INPUT
1
aba

SAMPLE OUTPUT
YES

15. Monk Teaches Palindrome

Monk introduces the concept of palindrome saying,"A palindrome is a sequence of characters which reads the same backward or forward."
Now, since he loves things to be binary, he asks you to find whether the given string is palindrome or not. If a given string is palindrome, you need to state that it is even palindrome (palindrome with even length) or odd palindrome (palindrome with odd length).

Input:
The first line consists of TT, denoting the number of test cases.
Next follow TT lines, each line consisting of a string of lowercase English alphabets.

Output:
For each string , you need to find whether it is palindrome or not.
If it is not a palindrome, print NO.
If it is a palindrome, print YES followed by a space; then print EVEN it is an even palindrome else print ODD.
Output for each string should be in a separate line.
See the sample output for clarification.

Constraints:
$1 \le T \le 50$
$1 \le$ length of string $\le 10^5$

SAMPLE INPUT
3
abc
abba
aba

SAMPLE OUTPUT
NO
YES EVEN
YES ODD

Explanation
The first string is not a palindrome.
The second and third strings are palindromes of even and odd lengths respectively.

16. The Strongest String - 1

Maga and **Alex** are good at string manipulation problems. Just now they have faced a problem related to string. But it is not a standard string problem. They have no idea to solve it. They need your help.
A string is called **unique** if all characters of string are distinct.
String s1s1 is called **subsequence** of string s2s2 if s1s1 can be produced from s2s2 by removing some characters of s2s2.
String s1s1 is **stronger** than s2s2 if s1s1 is lexicographically greater than s2s2.
You are given a string. Your task is to find **the strongest unique string** which is subsequence of given string.

Input:
first line contains length of string.
second line contains the string.

Output:
Output **the strongest unique string** which is subsequence of given string.

Constraints:
1≤|S|≤100000
All letters are lowercase English letters.

SAMPLE INPUT
5
abvzx

SAMPLE OUTPUT
zx

Explanation
Select all subsequence of the string and sort them in ascending order. The greatest of all is zx.

17. Ashish and Binary Matrix
Pulkit is really good at maths. Recently, he came to know about a problem on matrices. Amazed by the problem he got, he asked Ashish the same problem. Ashish also being good at maths solved the problem within 5 minutes. Now, its your time to solve the problem.
You will be given n*m binary matrix. You need to tell if it is possible to delete a column such that after deleting that column, rows of the matrix will be unique. If yes than print "**Yes**" else print "**No**".

[**Input**]
First line contains an integer t denoting no.of test cases.
Next line contains 2 integers n and m denoting no.of rows and columns.
Next n line contains binary string of length m each.

[**Output**]
For each test case output "**Yes**" or "**No**".

[**Constraints**]
1<=t<=100
1<=n<=1000
2<=m<=1000

SAMPLE INPUT
2
3 3
101
000
100
2 2
11
11

SAMPLE OUTPUT
Yes
No

18. Nobita and String
Doraemon gave Nobita a gadget that swaps words inside a string in the following manner :
If there are W words, word 1 is swapped with word W, word 2 is swapped with word W-1 and so on. The problem is that Nobita himself cannot verify the answer for large strings. Help him write a program to do so.

INPUT:
the first line of the input contains the number of test cases. Each test case consists of a single line containing the string.

OUTPUT:
output the string with the words swapped as stated above.

CONSTRAINTS:
|string length| <= 100000
string contains english alphabets and spaces

SAMPLE INPUT
1
hello world

SAMPLE OUTPUT
world hello

19. Divyansh's Love For Maths - A

Divyansh has got very good mathematical skills and he loves to play with numbers. He is very fond of making tricky mathematical questions. One day he gave three numbers to his friend **Atul** and asked him to find the **nth**number which can be formed by using 2, 3 and 5.
The starting numbers are: 2, 3, 5, 22, 23, 25, 32, 33, 35, 52...

Your Input: 769
NOTE You do not need to create a program for this problem you have to write your answers of given input in given code snippet

- To see how to submit solution please check this link

Compile and Test can give Wrong Answer. Once you have written your answer, press Submit button to check your result.

SAMPLE INPUT
999

SAMPLE OUTPUT
535335

20. Remove Duplicates

Given a string **S**. Your task is to remove all duplicates characters from the string **S**
NOTE:
1.) Order of characters in output string should be same as given in input string.
2.) String **S** contains only lowercase characters **['a'-'z']**.

input:
Input contain a single string **S**.

Output:
Print the string **S** with no any duplicate characters.

Constraints:
Test Files 1 to 5:
1<=**|S|**<=100

Test Files 6 to 10:
1<=**|S|**<=100000

Sample Input #1
hacker

Sample Output #1
hacker

Sample Input #2
hackerearth

Sample Output #2
hackert

Sample Input #3
programming

Sample Output #3
progamin

SAMPLE INPUT
iloveprogramming

SAMPLE OUTPUT
iloveprgamn

21. Sumit And Rohil

It's a fine sunny afternoon today in California. Looking at the pleasant weather, **Sumit** is all ready to go out and play with his friend Rohil. Unfortunately, **Rohil** is down with fever. Seeing that his friend is ill, Sumit decides not to go out - instead play with Rohil inside the house. Sumit loves math, on the contrary Rohil loves strings. Sumit decides to play a game that involves more of strings and less of Maths so that Rohil could be at ease and have fun playing it.

The game is simple and is played on a piece of paper. Sumit writes down a long list of names on that paper and passes it to Rohil. Rohil gets confused on seeing so many names on that paper and asks Sumit about the game. So, Sumit explains him the rules of the game. Rohil is supposed to partition the names into groups, such that:

- Each name belongs to exactly one group.
- Names that belong to the same group are pairwise anagrams.
- The first character of all the names in the same group are equal.
- The last character of all the names in the same group are equal.
- The number of groups is minimum possible.

Note: Two strings are called anagrams if it's possible to form one string from the other by changing the order of its characters.

Rohil would have won the game easily, if he would have been fit and fine but since he is ill right now he needs your help in winning the game. So, help out Rohil and do give him your blessings.

Input:
The first line contains a single integer **N** indicating the size of the list. This is followed by N lines where each line contains a name listed by Sumit.

Output:
In a single line print minimum number of groups in a partition that satisfy above conditions

Constraints:
1<= **N** <=100
1 <= **Length of a name** <= 100
All names will consist of lowercase English alphabets(a-z).

SAMPLE INPUT
6
vinay
vainy
vinit
viint
avinash
aasivnh

SAMPLE OUTPUT
3

Explanation
There are 3 special groups
1) vinay and vainy
2) vinit and viint
3) avinash and aavsinh

22. Name Game

John has recently learned about ASCII values. With his knowledge of ASCII values and character he has developed a special word and named it John's Magical word.
A word which consists of alphabets whose ASCII values is a prime number is a John's Magical word. An alphabet is john's Magical alphabet if its ASCII value is prime.
John's nature is to boast about the things he know or have learnt about. So just to defame his friends he gives few string to his friends and ask them to convert it to John's Magical word. None of his friends would like to get insulted. Help them to convert the given strings to John's Magical Word.
Rules for converting:
1.Each character should be replaced by the nearest John's Magical alphabet.
2.If the character is equidistant with 2 Magical alphabets. The one with lower ASCII value will be considered as its replacement.

Input format:
First line of input contains an integer T number of test cases. Each test case contains an integer N (denoting the length of the string) and a string S.

Output Format:
For each test case, print John's Magical Word in a new line.

Constraints:
1 <= T <= 100
1 <= |S| <= 500

SAMPLE INPUT
1
8
KINGKONG

SAMPLE OUTPUT
IIOGIOOG

23. String Division

Kevin has a string S consisting of N lowercase English letters.
Kevin wants to split it into 4 pairwise different non-empty parts. For example, string "happynewyear" can be splitted into "happy", "new", "ye" and "ar". He can't delete any characters or change the order of the characters.
Help Kevin and find if there exist at least one possible spliting.

Input format:
The first line of input will contain an integer **T**, denoting the number of test cases. Each of the next **T** lines contains a string **S**.

Output format:
For every test case output "YES" if it is possible to split the string and "NO" otherwise.

Constraints:

- 1 ≤ T ≤ 100
- 1 ≤ N ≤ 1000
- N ≤ 20 in test data worth 40% of all points

SAMPLE INPUT
2
ababca
aaabb

SAMPLE OUTPUT
YES
NO

24. Xenny and Partially Sorted Strings

Xenny had a list of **N** strings of equal length. He wanted to sort them by the first **M** characters only. That means, while sorting the list of strings, he only wanted to consider the first **M** characters of each string.
Help Xenny to find out the Kth string in the list after he sorts them.
Note: Xenny wanted to perform **stable sorting**.
Stable sorting algorithms maintain the relative order of records with equal keys (i.e. values). That is, a sorting algorithm is stable if

whenever there are two records R and S with the same key and with R appearing before S in the original list, R will appear before S in the sorted list.

Input

First line contains a single integer - **T**, which represents the number of testcases.
T testcases follow.
Each testcase is of the following format:
First line contains 3 space-separated integers - **N**, **K** and **M**.
N is the total number of strings Xenny has.
K is the index of the string in the list after sorting, which Xenny has to find.
M is the number of characters based on which sorting will be done by Xenny.
Then next **N** lines contain **N** strings (each line will contain one string) .

Output

For each testcase, print the **K**th string in the sorted list in a new line.

Constraints

1 ≤ T ≤ 50
1 ≤ N ≤ 10^3
1 ≤ Max Length of each String ≤ 10^3
1 ≤ M ≤ Max Length
M ≤ Max Length of each String ≤ 10^3

SAMPLE INPUT

1
3 1 3
abcdef
abcaaa
aabaaa

SAMPLE OUTPUT

aabaaa

Explanation

After performing sorting by the first **3** characters, the order is:
1. aabaaa
2. abcdef
3. abcaaa

25. Caesar's Cipher

Caesar's Cipher is a very famous encryption technique used in cryptography. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 33, DD would be replaced by GG, EE would become HH, XX would become AA and so on.
Encryption of a letter XX by a shift KK can be described mathematically as EK(X)=(X+K)EK(X)=(X+K) % 2626.
Given a plaintext and it's corresponding ciphertext, output the minimum non-negative value of shift that was used to encrypt the plaintext or else output −1−1 if it is not possible to obtain the given ciphertext from the given plaintext using Caesar's Cipher technique.

Input:

The first line of the input contains QQ, denoting the number of queries.
The next QQ lines contain two strings SS and TT consisting of only upper-case letters.

Output:

For each test-case, output a single non-negative integer denoting the minimum value of shift that was used to encrypt the plaintext or else print −1−1 if the answer doesn't exist.

Constraints:

- 1≤Q≤5
- 1≤|S|≤10^5
- 1≤|T|≤10^5
- |S|=|T|
-

SAMPLE INPUT

2
ABC

DEF
AAA
PQR

SAMPLE OUTPUT
3
-1

Explanation
In the first test case, A is replaced by D, B by E and C by F. It is easy to make out that a shift of 33 has been used for encrypting the plain text.
In the second test case, the value of the shift is not consistent for all letters of the plain text. Thus, we can safely come to the conclusion that the plain text has not been encrypted using Caesar's Cipher. Hence, the answer is no.

26. Mirror of Mahatma Gandhi

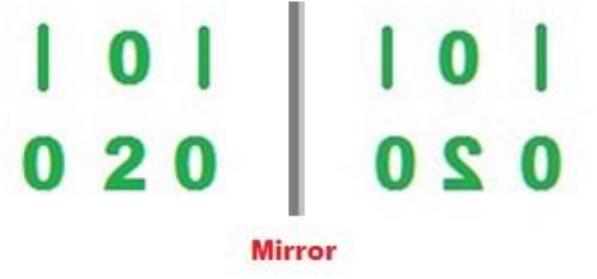
On the way to Dandi March, Gandhijee carried a mirror with himself.
When he reached Dandi, he decided to play a game with the tired people to give them some strength.
At each turn of the game he pointed out a person and told him to say a number **N**(possibly huge) of his choice.
The number was called lucky if that equals it's mirror image.

Input:
First line contains number of test cases **T**. Each test case contains a single integer **N**.

Output:
For each test case print "**YES**" if the number was lucky else print "**NO**" (quotes for clarity) in one line.

Constraints:
1 ≤ **T** ≤ 100
0 ≤ **N** ≤ 10^{100}

Image for Sample Test Cases :



SAMPLE INPUT
2
101
020

SAMPLE OUTPUT
YES
NO

Explanation
For 1st case, as clear from the image "101" and it's mirror image are identical. Hence, output "YES". For 2nd case, "020" and it's mirror image are not identical. Hence output "NO".

27. Sorted String

Little Ashish got a lot of *strings* as his birthday gift. He does not mind getting so many strings for free; in fact, he loves them. But, on noticing all the strings he received as a gift, Little Ashish, who's also a snob and a bit *OCD*kind of a guy, realizes that he does not like the way in which the strings are arranged.
He likes his strings sorted, in a different kind of a way. He wants his strings to be sorted based on the count of characters present in the string. For instance, if the string is: "**aaabbc**", then the desired string would be: **cbbaaa**. In case where the count of two characters is same, then the lexicographically smaller one will be printed first. For instance: "**aabbcc**" then, the output will be: "**aabbcc**".

Input:
First line of input contains number of test cases **T**. Each test case contains a single string **S**.

Output:
For each test cases print the sorted string.

Constraints:
1<=T<=100
1<=|S|<=100

Note:
String contains only lowercase characters ['a' to 'z'].

SAMPLE INPUT

```
3
aabbccdd
aabcc
hackerearth
```

SAMPLE OUTPUT

```
aabbccdd
baacc
cktaaeehhrr
```

28. Secret Messages!

X and Y are best friends and they love to chat with each other. But their recent concerns about the privacy of their messages has distant them. So they decided to encrypt their messages with a key, **K**, such that the character of their messages are now shifted **K** times towards right of their initial value. Their techniques only convert numbers and alphabets while leaving special characters as it is.
Provided the value **K** you are required to encrypt the messages using their idea of encryption.

INPUT FORMAT
The first line of the input contains, **T**, the number of messages. The next line contains **N**, and **K**, no of characters in the message and key for encryption. The next line contains the message.

OUTPUT FORMAT
Output the encrypted messages on a new line for all the test cases.

CONSTRAINS
1≤T≤100
1≤N≤10^6
0≤K≤10^6

SAMPLE INPUT

```
 2
12 4
Hello-World!
16 50
Aarambh@1800-hrs
```

SAMPLE OUTPUT

```
Lipps-Asvph!
Yypykzf@1800-fpq
```

Explanation
For the first test case, the message is "**Hello-World!**". Shifting "**H**" towards right by 4 we get "**L**", similarly sifting "**ello**" by 4 each we would get "**ipps**". Now since "**-**" is a special character we do not need to encrypt this character. After that we have "**W**" shifting "**W**" by 4 we come at "**A**", simlilarly shifting "**orld**" by 4 each we get "**svph**" and the last character is again a special character "**!**" so we do not encrypt this character.
For the second test case, we have **"Aarambh@1800-hrs"** we shift "**A**" by 50 which brings us to "**Y**" after 1 full circular rotation, the next character of the message is "**a**" which will be encrypted to "**y**", similarly "**rambh**" is encrypted in the same manner giving us "**pykz**". The next character is a special character "**@**", after that we have "**1**", shifting "**1**" by 50 we get "**1**" itself and similarly encrypting "**800**" gives us "**800**". The next character is again a special character "**-**", after which we have "**hrs**" which on encrypting will give is "**fpq**".

29. Decode

Given an encrypted message, Erwin encodes it the following way:
Removes the median letter of the word from the original word and appends it to the end of the encrypted word and repeats the process until there are no letters left.
A median letter in a word is the letter present in the middle of the word and if the word length is even, the median letter is the left one out of the two middle letters.
Given an encoded string, write a program to decode it.

Input Format:
The first line of input contains **T**, the number of test cases.
Each test case contains a String **S**, denoting the encoded word.

Output Format:
Print the decoded word for each test case in a separate line.

Constraints
1≤T≤100
1≤|S|≤100000

SAMPLE INPUT

```
2
wrien
reen
```

SAMPLE OUTPUT

```
erwin
eren
```

Explanation
In the first test case, Erwin encoded the String "erwin". At first, he wrote down the letter 'w' after which the string became "erin", he then wrote down 'r' and the remaining string was "ein", he then wrote 'i' and the string became "en" and so on he wrote down 'e' and 'n' to get the encoded string as "wrien".

30. Password

Danny has a possible list of passwords of Manny's facebook account. All passwords length is odd. But Danny knows that Manny is a big fan of palindromes. So, his password and reverse of his password both should be in the list.
You have to print the length of Manny's password and it's middle character.
Note : The solution will be unique.

INPUT
The first line of input contains the integer NN, the number of possible passwords.
Each of the following NN lines contains a single word, its length being an odd number greater than 22 and lesser than 1414. All characters are lowercase letters of the English alphabet.

OUTPUT
The first and only line of output must contain the length of the correct password and its central letter.

CONSTRAINTS
1≤N≤100

SAMPLE INPUT

```
4
abc
def
feg
cba
```

SAMPLE OUTPUT

```
3 b
```

31. Marut and Strings

Marut loves good strings. According to him, good strings are those which contain either all alphabets of **uppercase** or **lowercase**. While he is preparing for his exams, he finds many bad strings in his book and wants to convert them to good strings. But he wants to do this in minimum number of operations.
In one operation, he can pick only one character of any case and convert it to any other case.
As his exams are going on, he wants your help.

Input:
The first line contains an integer **T**, denoting the number of test cases.
Each test case consists of only one line with a string **S** which contains uppercase and lowercase alphabets..

Output:
Print the minimum number of operations, in which Marut can obtain a good string.
Print the answer for each test case in new line.

Constraints:
1 ≤ **T** ≤ 10
If **T** is not in this range, print "Invalid Test" (without the quotes)
1 ≤ Length of **S** ≤ 100
S can contain numbers, special characters but no spaces.
If the length of string is not in the above range or it does not contain **any** alphabets, print "Invalid Input" (without the quotes)
For Example, if the input is:
0
TestString
Print "Invalid Test" (without the quotes)

SAMPLE INPUT

```
3
abcEfg
!@6#2
123A
```

SAMPLE OUTPUT

```
1
Invalid Input
0
```

32. 47's Strings

Diana sent a message **"S"** to *agent 47* in an encrypted language so that no one else can decode it.
Now a string in encrypted language is said to be **valid** if and only if it contains the characters **'$' or '&'** . Agent 47 decided to count the number of valid sub-strings of the message **"S"** sent by Diana .
P.S. - '$' and '&' characters are considered as 4 -> '$' and 7 -> '&' in a key-board.
Can you help Agent 47 , in counting the **total no. of valid substrings** in encrypted message.
Note : Two sub-strings are different if they occur at different positions in the message.

Input :

- First line consists of a single integer **'T'** , i.e. no. of test cases .
- Each test case consists of a string **"S"**,i.e the message sent by Diana

Output :

- For each test case, output a single integer containing the count of total no. of valid sub-strings in new line.

Constraints :

- 1<=T<=10
- 1<=|S|<=10^6
- Message 'S' contains only lower case letters and characters '$' and '&'

SAMPLE INPUT

```
2
$bcd
flkjdh
```

SAMPLE OUTPUT
4
0

Explanation

- Sample test 1: For the first case, there are 4 sub-strings containing '$' .
- Sample test 2: There are no sub-strings having '$' or '&'.

33. Can you count?

You are given a string **s** consisting of lowercase English letters and/or '_' (underscore).
You have to **replace all underscores (if any) with vowels** present in the string.
The rule you follow is:
Each underscore can be replaced with any one of the vowel(s) that came before it.
You have to tell the total number of distinct strings we can generate following the above rule.

Input format:
The first line consists of an integer **t**, denoting the number of test cases.
Each of the next **t** lines consists of a string **s**.

Output format:
For each test case, output total number of distinct strings we can generate following the given rule.
Answer for each test case should come in a new line.

Input Constraints:
1≤t≤4000
1≤|s|≤10^5 |s| denotes string length.
Sum of |s||s| over all test-cases won't exceed 10^5.
It is guaranteed that there will be atleast **one** vowel before any '_' (underscore) character in the string.
Note:
It is guaranteed that answer won't go beyond 5∗10185∗1018.
SAMPLE INPUT
2
ab_ae_
abc

SAMPLE OUTPUT
2
1

Explanation
Test-case 1: First underscore can be replaced by only one vowel ('a'), while second underscore can be replaced by any one of the two vowels ('a' or 'e').
Distinct strings possible are: 1) abaaea 2) abaaee
So output is **2**.
Test-case 2: There are no '_' underscores. So we cannot do any replacement. So given string is the only answer. Output **1**.

34. Sumit's Task

One day Sumit's friend Himanshu got angry with him for not waking him in the morning, so he gave a tough task to Sumit to take the revenge.
"Given a string **S** consisting of letters **'O' and 'Z'**. Calculate the *minimum number of changes* required such that all **O's** are adjacent to each other and all **Z's** are adjacent to each other. One change is equivalent to swapping two neighbouring letters."
Sumit needs your help to complete this task as it is difficult for him to solve alone.

Input :

- The first line contains T - the number of test cases .
- Each test case consists of a string S consisting of characters 'O' and 'Z' only.

Output :

- Desired O/p

Constraints :

- 1<=T<=10

- 1<=|S|<= 10^6

SAMPLE INPUT
1
OOZOZO

SAMPLE OUTPUT
3

35. Palindromes
Everybody loves palindromes, but Artur doesn't.
He has a string **S** that consists of lowercase English letters ('a' - 'z'). Artur wants to find a substring **T** of **S** of the maximal length, so that **T isn't** a palindrome.

Input
The input contains a single line, containing string **S**. String **S** consists of lower case English letters.

Output
In a single line print the length of **T**

Constraints
1 ≤ **|S|** ≤ 100000

SAMPLE INPUT
aba

SAMPLE OUTPUT
2

Explanation
"aba" is a palindrome, but "ab" isn't.

36. Terrible Chandu
Chandu is a bad student. Once his teacher asked him to **print the reverse of a given string**. He took three hours to solve it. The teacher got agitated at Chandu and asked you the same question. Can you solve it?

Input:
The first line contains an integer **T, denoting the number of test cases.**
Each test case contains a **string S**, comprising of **only lower case letters.**

Output:
For each test case, print the **reverse of the string** S.

Constraints:
1 <= **T** <= 10
1 <= **|S|** <= 30

SAMPLE INPUT
2
ab
aba

SAMPLE OUTPUT
ba
aba

37. UpUp
You are given a string **S**. **S** consists of several words separated by one or more spaces. Word consists of Latin letters as well as other symbols (but not spaces).
In each word which starts from lowercase Latin letter replace starting letter with uppercase Latin letter.

Input
The only line contains **S**

Output
Output one line with modified string **S**.

Constraints
1 <= length of **S** <= 30 000

SAMPLE INPUT
Wish you were here

SAMPLE OUTPUT
Wish You Were Here

38. Count Enemies

Little Arjit is the leader of a marvellous fighting army. His team is very good at fighting against all their enemies. But like Hound from Game of Thrones, Little Arjit and his entire team is scared of fire. When they see fire, they feel threatened, and scared like a little child, and don't even mind giving up from a fight.
Now, you're given the location of the army men of Little Arjit, their enemies, and fire in a pathway. You've got to figure out how many enemies will survive after the massacre done by Little Arjit and his army.
Let's say Little Arjit and his army is denoted by 'X', all their enemies are denoted by 'O', and fire is denoted as '*'
Here's what happens:
When a X faces an O, i.e., when X is adjacent to O, X kills O. Always. Whenever there will be an **O** adjacent to **X** ,**X**will kill **O** . The moment X faces a *, he stops, and he cannot move any further.
Given a string containing X, O, * - can you help Little Arjit figure out how many of his enemies will still survive this attack because of fire?

Input:
First line of input contains a integer **T**,number of test cases. Next **T** lines contains a single string **S**.

Output:
For each test case print number of enemies survive after attack.

Constraints:
1<=**T**<=30
1<=**|S|**<=100000

SAMPLE INPUT
3
X*OO*XX
X*OX*XO*
X*OO*OO*X*OX*

SAMPLE OUTPUT
2
0
4

39. Lexical Analyzer

Alex has recently decided to learn about how to design compilers. As a first step he needs to find the number of different variables that are present in the given code.
So Alex will be provided NN statements each of which will be terminated by a semicolon(;). Now Alex needs to find the number of different variable names that are being present in the given statement. Any string which is present before the assignment operator denotes to a variable name.

Input Format: :
The first line contains a single integer NN.
Each of the next NN lines contains a single statement.
It is guaranteed that all the given statements shall be provided in a valid manner according to the format specified.

Output Format: :
Print the number of different variable name that are present in the given statements.

Constraints: :
1≤N≤10^5
1≤|Statement|≤100

SAMPLE INPUT
2
foo = 3;
bar = 4;

SAMPLE OUTPUT
2

Explanation
Foo and Bar are only two variables used inside the statements so answer is 2.

40. Faster than Flash

There is a meta human in Central City. The Flash has been given velocity-9 drug by this meta which reduces his speed. In order to defeat this meta Harrison wells has found String **S** and code.
Consider following code written by Harrison wells :

```
boolean function(String str){

    for(int i=0;i<str.length();i++){
      for(int j=i+1;j<str.length();j++){
        String sub=str.substring(i, j+1);
        // sub is substring of s from ith to jth character
        boolean flag=true;
        for(int k=0;k<sub.length();k++){
          if(sub.charAt(k)!=sub.charAt(sub.length()-k-1)){
            flag=false;
          }
        }
        if(flag) return false;
      }
    }
    return true;
  }
```

Now, Harrison Wells knows that the only way to defeat this meta is to find the longest substring of string **S** that has length greater than 1 and when given to the above function returns true.
Since this code is pretty slow and the Flash does not have enough speed to help central city. Help him to find the **longest substring** of the given input string which will return true when given to the above function. If there exists no such substring that has length greater than 1 then print **-1**.

Input
First line of input contains integer **T** , denoting number of test cases.
Each test case contains String **S**, which Harrison Wells has found.

Output
Print answer of each test case in new line.

Constrains
1<=T<=10
1<=|S|<=100000

SAMPLE INPUT
3
synapse
a
daiict

SAMPLE OUTPUT
7

-1
3

Explanation
In the first test case, ***synapse*** is the largest substring.
In the second test case, there is no substring has length greater then 1.
In the third test case, ***dai*** is the largest substring.

41. Don't Jump

Chandu and Kundu are bored of dramatic games around. As they are very good friends and scholars they decided to discover a new game.
In total they had **N** number of marbles with value inscribed on each of them from **1** to **N**.
Chandu being from royal family has a treasure box. He emptied the box for the game to proceed.
The game rule involved following operations :
- **Operation C** : Chandu places a marble in the box. Each different marble can move inside the box only once.
- **Operation K** : Kundu removes last entered marble from the box and notes down it's value on a chart.
They were deciding further rules when a third person Mr. Chirp comes as a spoiler of the game. Chirp gives them a number **N** which is the maximum value inscribed among all marbles, and a sequence **P**. **P** can possibly be empty. In that case, in input an empty line would be there.
Also, he added following **restriction** :
- Kundu can perform operation **K** to take out a marble **M** only when all the marbles from values **1** to **M-1** have went in the box before.
Mr. Chirp kept on chirping and told Chandu and Kundu to generate the sequence of **C**'s and **K**'s for the chart to get filled by pattern **P**.
Now, you have to help these kids before they jump out of their window in frustration.

Input Format :
First line contains a number **N**. Next line contains a pattern **P**(possibly empty).
Read till **end of file**.
Note: *Maximum input cases in a file doesn't exceed* 2*10^4.

Output Format :
For each case output on a separate line the **minimum length** pattern of **C**'s and **K**'s. Print **-1** if it's not possible to generate the pattern using the given operations.

Constraints :
$1 \le N \le 10^5$
$0 \le len(P) \le 100$
$1 \le P[i] \le N$

SAMPLE INPUT
41
6 11 8

SAMPLE OUTPUT
CCCCCCKCCCCCKKKK

Explanation
For getting 6 on chart first we need six C Operations for 1,2,3,4,5 and 6 followed by one K operation for getting 6 out. Afterwards for 11 we perform five C operations for 7,8,9,10,11 followed by one K operation. And finally, for 8 we need four K operations for 10, 9, 8 respectively.

42. Short Name

You are given a string *S*. You need to print short name of the given string.

Input Format
The first and only line of input contains a string *S*.

Output Format
You need to print the short name of provided string.

SAMPLE INPUT
Mohd Kafeel Khan

SAMPLE OUTPUT
M. K. Khan

43. Jumble Letter

Prashant started to attend programming lessons. On the first lesson his task was to write a simple program. The program was supposed to do the following: in the given string, consisting if uppercase and lowercase Latin letters, it:

- deletes all the vowels,
- inserts a character "." before each consonant,
- replaces all uppercase consonants with corresponding lowercase ones.

Vowels are letters "A", "O", "Y", "E", "U", "I", and the rest are consonants. The program's input is exactly one string, it should return the output as a single string, resulting after the program's processing the initial string.
Help Prashant cope with this easy task.
Input First line of input contains an integer T number of test cases. The next T line represents input string of Prashant's program. This string only consists of uppercase and lowercase Latin letters and its length is from 1 to 100, inclusive.
Output Print the resulting string. It is guaranteed that this string is not empty.

SAMPLE INPUT
1
odn

SAMPLE OUTPUT
.d.n

44. Invert Case of Character

You are given a string *S* followed by two integers *N1* and *N2*. You need to change the case of character at those two indices as specified by the two integers.

Input Format
First line contains the string *S* Second line contains two integers separated by single space

Output Format
It should print the string with case changed at those locations

SAMPLE INPUT
Jamia Hamdard
2 9

SAMPLE OUTPUT
JAmia HaMdard

45. The Cheapest Palindrome

Palindrome is a string that reads the same backward as forward, e.g., madam.
You are given a string whose length is even, and each character of the string is either 'a', 'b' or '/' Your task is to replace each occurrence of '/' in string with either 'a' or 'b' such that the string becomes a palindrome.
You are also given two integers, aCost and bCost. Replacing '/' with 'a' costs aCost, and replacing '/' with 'b' costs bCost.
Return the minimum cost of replacing '/' by 'a' and 'b' such that the string turns into a palindrome. If it is impossible to obtain a palindrome, return -1.

Constraints

- string will contain between 1 and 10000 characters, inclusive.
- The length of string will be even.
- Each character of the string will be either 'a' or 'b' or '/'.
- aCost will be between 1 and 100, inclusive.
- bCost will be between 1 and 100, inclusive.

Input Format
First line of input will contain the number of test cases. For each test case, there will be of three lines, the first line is the string whose palindrome is to be constructed, the second is the aCost and the third is the bCost

Examples
1

aba/bab/
4
6
Returns: 10
The only way to produce a palindrome is to replace 4th character of string with 'b' and 8th character with 'a'. The first replacement costs 4, the second costs 6, so the total cost is 4+6=10.
aaaabbbb
12
34
Returns: -1
There is no '/' character, and string is not a palindrome. We have no way to change it into a palindrome.

SAMPLE INPUT
1
baba//aaa/ab//
72
23

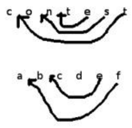
SAMPLE OUTPUT
213

46. Swapping Game

Sherlock and Watson are playing swapping game. Watson gives to Sherlock a string **S** on which he has performed **K** swaps. You need to help Sherlock in finding the original string.
One swap on a string is performed in this way:

- Assuming 1 indexing, the **i'th** letter from the end is inserted between **i'th** and **(i+1)'th** letter from the starting.

For example, we have "contest". After one swap, it would change to "ctosnet". Check this image:



Input:
First line contains **K**, the number of swaps performed by Watson. Next line contains **S**, the string Watson gives to Sherlock.

Output:
You have to print in one line the original string with which Watson had started.

Constraints:
1 ≤ K ≤ 10^9
3 ≤ Length(S) ≤ 1000
All characters in S will be small English alphabets.

SAMPLE INPUT
3
hrrkhceaate

SAMPLE OUTPUT
hackerearth

Explanation
When swapping is done on "hackerearth" 3 times, it transforms to "hrrkhceaate".

47. Divyansh's Love For Maths - B

Divyansh has got very good mathematical skills and he loves to play with numbers. He is very fond of making tricky mathematical questions. One day he gave three numbers to his friend **Atul** and asked him to find the **nth**number which can be formed by using 2, 3 and 5.
The starting numbers are: 2, 3, 5, 22, 23, 25, 32, 33, 35, 52...
Your Input:1000000000000000000
NOTE You do not need to create a program for this problem you have to write your answers of given input in given code snippet

- To see how to submit solution please check this link

Compile and Test can give Wrong Answer. Once you have written your answer, press Submit button to check your result.

SAMPLE INPUT

```
20
```

SAMPLE OUTPUT

```
253
```

48. Find the substrings

Given a string **S** made of letters **a**, **b** and **c**, find the number of sub strings that do not contain all the letters **a**, **b**and **c**. That is the number of sub strings that do not contain at least one of the letters **a** or **b** or **c**.
Note that the sub string should contain atleast one letter, that is it should not be empty string.

Input
First line of the input is the number of test cases **T**. It is followed by **T** lines. Each test case is a single line which is the string **S**.

Output
For each test case, print a single number, the required answer.

Constraints
1 <= |S| <= 10^6

SAMPLE INPUT

```
3
ababa
abc
babac
```

SAMPLE OUTPUT

```
15
5
12
```

49. String Queries

Given a 11-indexed String SS of length NN, consisting of lowercase English alphabets, you need to answer queries based on the properties of this string.
You shall be given QQ queries to answer. In each query you shall be given 22 integers LL and RR. Now , you need to report the minimum number of String elements you need to delete in the substring of String SS ranging from LL to RR inclusive, so that each distinct character occurring in the range occurs equal number of times.
For example, consider the String "abcdab""abcdab" . Here, the characters occurring in this String are 'a''a'. 'b''b', 'c''c' and 'd''d'. On deleting one occurrence of 'a''a' as well as 'b''b' one of the possible Strings is abcdabcd. Each character occuring in the range now occurs equal number of times. So, the minimum number of deletions is two.
Note that we need to equalize the count of only characters that occur in the range, and not of characters that do not. It is allowed to delete all occurences of a character, so that it no longer occurs in the range at all
Can you do it ?

Input Format :
The first line contains 22 space separated integers NN and QQ. The next line contains the String SS. Each of the next QQ lines contains 22 space separated integers LL and RR, denoting the parameters of the ithith query.

Output Format :
Print the answer to each query on a new line.

Constraints :
$1 \leq N, Q \leq 10^5$
$1 \leq L \leq R \leq N$

SAMPLE INPUT

```
8 2
abcdabcd
1 6
2 7
```

SAMPLE OUTPUT

2
2

Explanation
The explanation to the 1st1st query in the sample has been provided in the problem statement. For the 2nd2nd query, the substring required here is "bcdabc""bcdabc" . We can delete from this substring one occurence one occurence of ′b′′b′and ′c′′c′ to achieve the desired result. Thus, the answer is 22.

50. Missing Alphabets

Adriana was playing with the English alphabet. When she was done playing with the alphabet, she realised that she had jumbled up the positions of the letters. Now, given a set of words, she wondered what would be the dictionary ordering of these words based on the new alphabet ordering which she made.
In other words, given a permutation of the English alphabet, E and a set of words S, you need to output the lexicographical ordering of the words in the set S based on the new alphabet, E.

Input
The first line will contain a single integer, T, denoting the number of test cases. T lines follow.
For each test case:
The first line will contain a string, E, the new alphabet ordering, which will be a permutation of 'abcdefghijklmnopqrstuvwxyz'
The next line will contain a single integer M, the size of the set S. S lines follow, each containing a single word, containing lowercase latin characters.

Output
For each test case, output S lines, each line containing one word from the set S, ordered lexicographically.

Constraints
1 <= T <= 1000
1 <= M <= 100
1 <= |W| <= 50

SAMPLE INPUT
```
2
abcdefghijklmnopqrstuvwxyz
2
aa
bb
bacdefghijklmnopqrstuvwxyz
2
aa
bb
```

SAMPLE OUTPUT
```
aa
bb
bb
aa
```

51. Permutation

You are given a string S. You need to print all possible permutation of that string.

Input Format
First and only line contains a string whose permutations need to be displayed

Output Format
All possible permutations should be displayed in single line with single spaces between them

SAMPLE INPUT
```
abc
```
SAMPLE OUTPUT
```
abc acb bac bca cab cba
```

52. Little Vihu and Special Strings

Little Vihu loves special strings. According to him, special strings are those which have largest **P(s)** value. **P(s)** is defined as follows for a *string* '*s*' of *length* ***n*** with *0-based indexing-->*
P(s) = ∑n−1i=0C(s[i])∗(2−i)∑i=0n−1C(s[i])∗(2−i)
where **C(x)** is value of that character. Value of a to z is 1-26 ie. a=1, b=2,c=3.......so on upto....z=26.
Now, to make a string s a special string, he can break string into two parts of any size and swap their positions.Your task is to find the string having **maximum P(s) value** which can be formed using above operation.

Example

s="avika"

The above string can be partioned in these following ways and resulting string will be-->

Partioned Strings	Resultant String
a + vika	vikaa
av + ika	ikaav
avi + ka	kaavi
avik + a	aavik
avika +	avika

Now, it can be clearly seen as "vikaa" has maximum value, therefore "vikaa " is the answer.

Input

Single line containing a string consisting only of lowercase alphabets.

output

Single line containing the string with max P(s).

Constraints

1<=n<=15

Scoring-->

count=Number of characters
count < 75 --> *100
*count < 150 --> *100 - (0.1 * count)
*count < 480 --> *100 - (0.2 * count)
*count >= 480 --> *10*

SAMPLE INPUT

avika

SAMPLE OUTPUT

vikaa