**Non Repeating Character**

[ad-hoc](http://www.practice.geeksforgeeks.org/tag-page.php?tag=ad-hoc&isCmp=0)[string](http://www.practice.geeksforgeeks.org/tag-page.php?tag=string&isCmp=0)[Amazon](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Amazon&isCmp=1)

Given a string s consisting of lowercase Latin Letters, find the first non repeating character in s.

**Input:**

The first line contains T denoting the number of testcases. Then follows description of testcases.  
Each case begins with a single integer N denoting the length of string. The next line contains the string s.  
   
**Output:**

 For each testcase, print the first non repeating character present in string.  
   
 Print -1 if there is no non repeating character.  
   
**Constraints:**

 1<=T<=30  
 1<=N<=100

**Example:**

Input :

3  
5    
hello  
12  
zxvczbtxyzvy  
6  
xxyyzz

Output :

h  
c  
-1

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=196>

#include <iostream>

#include <stdio.h>

#include <math.h>

#include <map>

#include <conio.h>

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int N;

scanf("%d", &N);

std::string S;

cin >> S;

std::map<char, int> map;

for(int i =0; i < N; i++) {

map[S[i]]++;

}

bool encontrado=false;

for(int i =0; i < N; i++) {

if(map[S[i]] == 1) {

printf("%c\n", S[i]);

encontrado = true;

break;

}

}

if(!encontrado){

printf("%d\n", -1);

}

}

getch();

return 0;

}