pb Password

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 4MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Descripción**

Para que un password sea seguro debe seguir las siguientes reglas:

* Tener al menos una vocal.
* No tener dos o más caracteres consecutivos iguales.
* Tener una longitud de al menos 12 caracteres.
* Tener al menos un carácter numérico.

**Problema**

Escribe un que nos ayude a determinar si un password es seguro o no.

**Entrada**

En la primera línea un número n indicando la cantidad de caracteres que conforman el password, donde 1≤n≤100, en la siguiente línea n caracteres que pueden ser letras de la A a la Z, números del 1 al 9. las letras minúsculas y mayúsculas se consideran diferentes.

**Salida**

Si el password es seguro de acuerdo a las reglas establecidas debes escribir la palabra SEGURO en caso contrario debes escribir la frase INSEGURO.

**Consideraciones**

Tu programa se evaluará con varios casos de prueba.

**Ejemplos**

| **Entrada** | **Salida** |
| --- | --- |
| 5  abcdd | INSEGURO |
| 12  aft156YTU512 | SEGURO |

*Fuente: Curso de Profesores*

Problema subido por: [CursoProfesores](https://omegaup.com/profile/CursoProfesores/)

| **ID** | **Status** | **Porcentaje** | **Penalty** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/pb-Password#problems/new-run) | | | | | | | | |
| e2c829cf | Respuesta parcialmente correcta | 57.14% | 0 | java | 17.80 | 2.53 |  |  |
| 056840bd | Respuesta parcialmente correcta | 57.14% | 0 | java | 16.79 | 2.03 |  |  |
| afec84cb | Respuesta correcta | 100.00% | 0 | java | 16.94 | 1.94 |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/pb-Password#problems>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication88;

import java.util.Scanner;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication88 {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// TODO code application logic here

Scanner sc = new Scanner(System.in);

int n = Integer.parseInt(sc.nextLine());

String s;

s = sc.nextLine();

if(n < 12) {

System.out.println("INSEGURO");

} else if(n >= 12) {

String answer = "INSEGURO";

boolean vocal = false;

boolean consec = false;

boolean numero = false;

String vocales = "aeiouAEIOU";

String numeros = "0123456789";

for(int i = 0; i + 1 < n; i++) {

if(vocales.contains( String.valueOf( s.charAt(i))) ){

vocal = true;

}

if(numeros.contains(String.valueOf(s.charAt(i)))) {

numero = true;

}

if(s.charAt(i) == s.charAt(i+1)){

consec = true;

}

}

if(vocales.contains( String.valueOf( s.charAt(s.length()-1))) ){

vocal = true;

}

if(numeros.contains(String.valueOf(s.charAt(s.length()-1)))) {

numero = true;

}

if(vocal && numero && !consec) {

answer = "SEGURO";

}

System.out.println(answer);

}

}

}