

# Reglas:

---

- 1.- Tiene que resolverlos solo.
- 2.- No más de 48 horas para terminar este reto.
- 3.- Enviar el código fuente sin archivos binarios, ejecutables, etc. ó mediante github/ gitlab.
- 4.- Puede consultar todo aspecto técnico en google.
- 5.- Resolver los dos problemas en el lenguaje de mayor dominio (más seguro) y al menos uno en un lenguaje que desees aprender.

## Problema 1

---

### Descripción

Existe una técnica de encriptación que ocupa una agencia para poder enviar instrucciones a sus agentes. Para enviar una instrucción, la agencia transmite un mensaje donde la instrucción aparece entre otros caracteres. Por ejemplo la instrucción *CeseAlFuego* puede ser enviada como *XcamakCeseAlFuegoDLKmN*. Al recibir el mensaje, los agentes (con la ayuda de un libro con todas las instrucciones posibles) determinan cual es la instrucción escondida en el mensaje. Máximo existe una instrucción escondida por mensaje aunque es posible que no haya ninguna instrucción en el mensaje.

Desafortunadamente el transmisor que ocupan para el envío de los mensajes tiene una falla. En lugar de enviar los caracteres una sola vez, esta enviándolos una, dos o hasta tres veces. Por ejemplo, el mensaje anterior pudiera ser enviado así:

*XXcaaamakCCesseAAIlFueeegooDLLKmmNNN*. Esto hace que sea más difícil para los agentes el encontrar una instrucción. (Nota: Ninguna instrucción en el libro de instrucciones contiene dos letras iguales seguidas)

El programa recibe dos instrucciones y un mensaje, y el resultado debe ser si existe o no una instrucción escondida en el mensaje.

### Formato de Entrada

- La entrada al programa es un archivo de texto que consiste en cuatro líneas.

- La primera línea son tres enteros  $M_1$ ,  $M_2$  y  $N$ .  $M_1$  y  $M_2$  es el número de caracteres de las dos instrucciones y  $N$  es el número de caracteres en el mensaje.
  - $N$  siempre estará entre 3 y 5000 inclusive
  - $M_1$  y  $M_2$  siempre estarán entre 2 y 50 inclusive
- La segunda línea contiene la primera instrucción
- La tercera línea contiene la segunda instrucción
- La cuarta línea contiene el mensaje
  - Los caracteres posibles en el mensaje son [a-zA-Z0-9]

## Formato de Salida

- La salida del programa es un archivo que contiene dos líneas.
  - La primera línea contiene un SI si la primer instrucción se encuentra escondida en el mensaje o un NO de lo contrario
  - La segunda línea contiene un SI si la segunda instrucción se encuentra escondida en el mensaje o un NO de lo contrario

## Ejemplo

### Entrada

```
11 15 38
CeseAlFuego
CorranACubierto
XXcaaamakCCessseAAIfueeegooDLLKmmNNN
```

### Salida

```
SI
NO
```

## Problema 2

---

### Descripción

Existe un juego en el que hay dos jugadores y varias rondas en los que cada jugador obtiene una puntuación. Al final de cada ronda se observa el acumulado de la puntuación y la diferencia. Al terminar el juego quien gana es aquel que haya conseguido la mayor ventaja

Por ejemplo esto pudiera ser el marcador de 5 rondas:

Ronda	Jugador 1	Jugador 2
1	140	82
2	89	134
3	90	110
4	112	106
5	88	90

Este es un ejemplo con el marcador acumulado al final de cada ronda y la ventaja obtenida.

Ronda	Jugador 1	Jugador 2	Líder	Ventaja
-------	-----------	-----------	-------	---------

1	140	82	Jugador 1	58
2	229	216	Jugador 1	13
3	319	326	Jugador 2	7
4	431	432	Jugador 2	1
5	519	522	Jugador 2	3

El ganador en este caso sería el Jugador 1 ya que el obtuvo la máxima ventaja (58) al final de la primera ronda durante el juego.

El programa recibe los marcadores y deberá indicar el ganador (no hay empates, se puede asumir siempre existe un ganador único)

## Formato de Entrada

- La entrada al programa es un archivo de texto que consiste en varias líneas.
  - La primer línea es un entero menor o igual a 10000 indicando el número de rondas
  - Después hay una línea por ronda con los marcadores de los dos jugadores

## Formato de Salida

- La salida del programa es un archivo que contiene una línea
  - Esta línea contiene dos enteros, el primer entero es 1 o 2 indicando quien fue el ganador, y el segundo entero es la ventaja con la que gana ese jugador

## Ejemplo

### Entrada

```
5
140 82
89 134
90 110
```

112 106  
88 90

**Salida**

1 58