

# OCI PUC - 5

Julio 2016

## A. El juego de los bloques - PALL01

Los habitantes de Byteland regularmente juegan un juego: Tienen bloques, cada uno indicando un entero entre 0 y 9. Los bloques son ordenados de manera aleatoria sin mirar para formar distintos números, teniendo cuidado de que el primer bloque nunca sea 0. Una vez que forman un número lo leen en orden inverso para verificar que el número y su reverso sean el mismo. Si ambos números son el mismo, el jugador gana. Llamamos a tales números *palíndromo*

Ash justo vio este juego y quiere simularlo en un computador. Como primer paso, quiere recibir un input del usuario, verificar si es un palíndromo y declara si le usuario gana o no.

### Input

La primera línea del input contiene **T**, el número de casos de prueba. Esto es seguido por **T** líneas que contienen un entero **N**

### Output

Para cada input imprime "wins" si el número es palíndromo y "losses" si no.

### Restricciones

- $1 \leq T \leq 20$
- $1 \leq N \leq 10000$

### Ejemplo

Input :

3  
331  
666  
343

Output:  
losses  
wins  
wins

## B. Chef y Dos Strings - CHEFSTLT

El Chef ha encontrado dos hojas de papel muy viejas, cada una de las cuales originalmente contenía un string en minúscula con letras latinas. Los strings en ambas hojas tienen el mismo largo. Sin embargo, dado que las hojas son muy viejas, algunas letras se han vuelto ilegibles.

Al Chef le gustaría estimar la *diferencia* entre los strings. Asumamos que el primer string es llamado **S1** y el segundo **S2**. Los símbolos ilegibles son especificados con el símbolo de pregunta '?'. La *diferencia* entre los strings es igual al número de posiciones  $i$ , tales que  $S1_i$  no es igual a  $S2_i$ , donde  $S1_i$  y  $S2_i$  representan el símbolo en la posición  $i$  en **S1** y **S2** respectivamente.

Al Chef le gustaría saber la diferencia mínima y la máxima entre los dos strings, si cambia todos los símbolos ilegibles por letras latinas en minúscula. Ahora, conociendo la expertis del Chef programando, debes haber adivinado que él necesita tu ayuda para resolver este problema. ¡Ve y ayúdalo!

### Input

La primea línea del input contiene un entero **T** que indica el número de casos de prueba. La descripción de los **T** casos son como sigue:

La primera línea de un caso de prueba contiene un string **S1** La segunda línea de un caso de prueba contiene un string **S2** Ambos strings consiste en letras latinas en minúscula y signos de interrogación donde los símbolos son ilegibles.

### Output

Por cada caso de prueba, imprime la mínima y máxima diferencia entre dos strings dados separada por un único espacio.

### Restricciones

- $1 \leq T \leq 100$
- $1 \leq |S1|, |S2| \leq 100$
- Subtarea 1 (25 puntos):  $|S1| = 1$
- Subtarea 2 (10 puntos): ni **S1** ni **S2** contienen símbolos ilegibles
- Subtarea 3 (65 puntos):  $1 \leq |S1|, |S2| \leq 100$

### Ejemplo

Input :

3

a?c

??b

???a  
???a  
?abac  
aba?w

Output:  
1 3  
0 3  
3 5

## Explicación

**Caso de ejemplo 1.** Puedes cambiar los signos de interrogación en los strings de forma de obtener  $\mathbf{S1} = abc$  y  $\mathbf{S2} = abb$ . En ese caso  $\mathbf{S1}$  y  $\mathbf{S2}$  diferirán en una posición. Por el otro lado, puedes cambiar las letras para que  $\mathbf{S1} = abc$  y  $\mathbf{S2} = bab$ . En ese caso, los strings diferirán en las tres posiciones.

**Caso de ejemplo 2.** Cambia los signos de interrogación de esta manera:  $\mathbf{S1} = dcba$ ,  $\mathbf{S2} = dcba$ , entonces los strings diferirán en **0** posiciones. También puedes cambiar los signos de interrogación de modo que  $\mathbf{S1} = aaaa$ ,  $\mathbf{S2} = dcba$ , entonces los strings diferirán en **3** posiciones.

**Caso de ejemplo 3.** Cambia los signos de interrogación de esta manera:  $\mathbf{S1} = aabac$ ,  $\mathbf{S2} = abaaw$ , entonces los strings diferirán en **3** posiciones. Luego, cambia los signos de interrogación de la siguiente manera:  $\mathbf{S1} = xabac$ ,  $\mathbf{S2} = abayw$ , entonces los strings diferirán en **5** posiciones.

## C. Lapíndromos - LAPIN

*Lapíndromo* se define como un string tal que cuando se parte por la mitad, entrega dos mitades que tienen los mismos caracteres y la misma frecuencia para cada carácter. Si hay un número impar de caracteres entonces se ignora el carácter de al medio. Por ejemplo **gaga** es un lapíndromo, ya que las dos mitades **ga** y **ga** tienen los mismos caracteres con la misma frecuencia. También, **abccab**, **rotor** y **xyzxy** son unos pocos ejemplos de lapíndromos. Nota que **abbaab** NO es lapíndromo. Las 2 mitades contienen los mismos caracteres pero su frecuencia no calza. Tu tarea es simple. Dado un string, necesitas indicar si es un lapíndromo.

### Input

La primera línea del input contiene un único entero **T**, el número de casos de prueba. Cada caso de prueba contiene un string **S** compuesto solo de letras del alfabeto inglés in minúscula.

### Output

Para cada caso de prueba, escribe en una línea separada: "YES" si el string es lapíndromo y "NO" si no.

### Restricciones

- $1 \leq T \leq 100$
- $2 \leq |S| \leq 1000$ , donde  $|S|$  denota el largo de **S**

### Ejemplo

Input:

```
6
gaga
abcde
rotor
xyzxy
abbaab
ababc
```

Output:

```
YES
NO
YES
```

YES  
NO  
NO