

## ***Semana 12 Python***

L12A\_16-1

[Problema](#)

[Entrada](#)

[Salida](#)

L12B\_16-1

[Problema](#)

[Entrada](#)

[Salida](#)

L12C\_16-1

[Problema](#)

[Entrada](#)

[Salida](#)

## Semana 12 Python

### L12A\_16-1

Tiempo límite: 1 segundo

## Problema

El Longest common subsequence problem (Problema de subsecuencia común más larga) también conocido como LCS problem, se trata de encontrar una subsecuencia más larga que es común en un conjunto de secuencias ( En la mayoría de los casos 2 secuencias ).

Para este problema usted debe encontrar todas las subsecuencias de longitud máxima para 2 cadenas. Tenga en cuenta que si una cadena de longitud máxima aparece más de una vez esta no se debe imprimir nuevamente.

## Entrada

La primera línea consiste en un entero  $t$  el número de casos de prueba, cada caso de prueba está compuesto por 2 líneas cada una compuesta por una cadena de no más de 30 caracteres.

## Restricciones / Consideraciones

$$0 < t < 100$$

## Salida

Por cada caso de prueba se debe imprimir todas las subsecuencias máximas cada una en una línea. Las cadenas deben ser impresas en orden lexicográfico.

Ejemplo de Entrada	Ejemplo de Salida
2 abcbdad bdcaba aba bab	Case #1: bcab bcba bdab Case #2: ab ba

## Semana 12 Python

### L12B\_16-1

Tiempo límite de ejecución: 2 segundos

## Problema

Fabian y Felipe son monitores de algoritmos en una Universidad muy muy lejana y sus profesores les han encargado detectar copia en los exámenes finales. El profesor Yoan les ha sugerido que utilicen la siguiente métrica entre dos códigos A y B:

Código A:

```
def punto1(graph,source,target): random.randint(0,len(graph)) * target
```

Código B:

```
def kha(graph,zource,TargEt): random.randint(0,len(graph)) * (target/1.0)
```

Para ello el profesor Yoan les dice “Encuentren dos pedazos *contiguos* de código que sean exactamente iguales, para todos los pedazos posibles seleccionen el de tamaño máximo”. Dicho valor se expresa como  $\delta(A,B)$ . En el caso anterior el valor  $\delta(A,B) = 35$ . Tenga en cuenta que en python los espacios son muy importantes, por lo tanto la *longitud* y la *comparación* debe tener en cuenta exactamente los espacios.

El profesor Yoan se encuentra en unas merecidas vacaciones, y los monitores son demasiado perezosos como para implementar un programa antiplagio... ¿Puede usted ayudarlos?

## Entrada

La entrada comienza con un número  $T$  que indica el número de casos de prueba. Cada caso de prueba contiene 2 líneas que representan el código A y el código B.

## Restricciones/Consideraciones

$$1 \leq T \leq 10$$

$$1 \leq |A|, |B| \leq 1000$$

## Salida

Por cada caso de prueba debe imprimir en una sola línea el valor de  $\delta(A,B)$ .

## Semana 12 Python

Ejemplo de Entrada	Ejemplo de Salida
<pre>3 def punto1(graph,source,target): random.randint(0,len(graph)) * target def kha(graph,zource,TargEt): random.randint(0,len(graph)) * (target/1.0) a = b + 3; c = len(xd) a = b + 3; c = len(xd) f(n) = f(n-1) + f(n-2) f(n) = f( n - 2 ) + f(n-1)</pre>	<pre>35 22 9</pre>

## Semana 12 Python

### L12C\_16-1

Tiempo límite: 13 segundos

#### Problema

Fabio es el hermano mayor de Germán, su mamá Helga les dice que deben ir a comprar los alimentos necesarios para hacer el almuerzo, los dos hermanos a regañadientes deciden ir a la tienda. Sin embargo para divertirse deciden ir a las maquinitas de la tienda de don Dave utilizando las vueltas del mandado, sin embargo don Dave les dio un billete de **N** pesos y deben cambiarlo en la tienda de don Yoan. Al llegar a donde don Yoan el decide ayudar a los muchachos cambiando el billete (Sin embargo, don Yoan sabe que no puede quedarse sin monedas) debido a que en sus tiempos libres le gustan las matemáticas y la programación. Don Yoan decide cambiar el billete de **N** pesos con la **menor** cantidad de monedas posible. Debido a que don Yoan está ocupado resolviendo problemas de Strings, decide pedir su ayuda para calcular la mínima cantidad de monedas que se requieren para cambiar el billete dadas las denominaciones de las monedas que posee.

*Nota 1: Don Yoan posee infinitas monedas de cada denominación.*

*Nota 2: La denominación de cada moneda es el valor en pesos de la misma, por ejemplo 50 pesos es la denominación para una moneda de 50 pesos.*

**Si su solución es recursiva recuerde incluir esta línea antes de leer los datos `sys.setrecursionlimit(100000000)`.**

#### Entrada

La primera línea consiste en un entero  $t$  el número de casos de prueba.

Cada caso de prueba comienza con un entero  $n$  y un entero  $m$  separados por un espacio,  $n$  indica la denominación del billete a cambiar, en la siguiente línea hay  $m$  enteros separados ( $s_m$ ) por un espacio que representan las denominaciones de las monedas.

## Semana 12 Python

### Restricciones/Consideraciones

$$1 \leq n \leq 1000$$

$$1 \leq m \leq 1000$$

$$1 \leq s_m \leq 1000$$

### Salida

Por cada caso de prueba se debe imprimir la cantidad mínima de monedas a utilizar. Si no es posible hacer el cambio de billete con las monedas dadas debe imprimir -1.

Ejemplo de Entrada	Ejemplo de Salida
2 30 3 25 10 5 11 4 9 6 5 1	2 2