

## 1. Código

```
class lcs:
    def calcularMatriz(self, x, y):
        m = len(x)
        n = len(y)
        c = [[0 for j in range(n+1)] for i in range(m+1)]
        for i in range(1, m+1):
            for j in range(1, n+1):
                if x[i-1] == y[j-1]:
                    c[i][j] = c[i-1][j-1] + 1
                else:
                    c[i][j] = max(c[i-1][j], c[i][j-1])
        return c
    def reconstruirLcs(self, x, y, c):
        i = len(x)
        j = len(y)
        lcs = ""
        while i > 0 and j > 0:
            if x[i-1] == y[j-1]:
                lcs = x[i-1] + lcs
                i -= 1
                j -= 1
            elif c[i-1][j] >= c[i][j-1]:
                i -= 1
            else:
                j -= 1
        return lcs

    def identificarSolucionesOptimas(self, x, y, c, i, j):
        if i == 0 or j == 0:
            return [""]
        if x[i-1] == y[j-1]:
            soluciones = self.identificarSolucionesOptimas(x, y, c, i-1, j-1)
            return [solucion + x[i-1] for solucion in soluciones]
        if c[i-1][j] > c[i][j-1]:
            return self.identificarSolucionesOptimas(x, y, c, i-1, j)
        if c[i][j-1] > c[i-1][j]:
            return self.identificarSolucionesOptimas(x, y, c, i, j-1)
        arriba = self.identificarSolucionesOptimas(x, y, c, i-1, j)
        izquierda = self.identificarSolucionesOptimas(x, y, c, i, j-1)
        return arriba + izquierda

    def presentarSolucionesOptimas(self, x, y, c):
        soluciones = self.identificarSolucionesOptimas(x, y, c, len(x), len(y))
        print("Soluciones óptimas:")
        print("*****")
        for solucion in soluciones:
            print(solucion)

clase = lcs()
x = "SPRINGTIME"
y = "PRINTING"
c = clase.calcularMatriz(x, y)
lcs = clase.reconstruirLcs(x, y, c)
print("*****")
print("Matriz:")
for i in c:
    print(i)
print("*****")
print("La subsecuencia común más larga es:", lcs)
print("Su longitud es:", len(lcs))
print("*****")
clase.presentarSolucionesOptimas(x, y, c)
```

## 2. Terminal

```
01107/Complejidad/LCS.py
*****
Matriz:
[0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 1, 1, 1, 1, 1, 1, 1, 1]
[0, 1, 2, 2, 2, 2, 2, 2, 2]
[0, 1, 2, 3, 3, 3, 3, 3, 3]
[0, 1, 2, 3, 4, 4, 4, 4, 4]
[0, 1, 2, 3, 4, 4, 4, 4, 5]
[0, 1, 2, 3, 4, 5, 5, 5, 5]
[0, 1, 2, 3, 4, 5, 6, 6, 6]
[0, 1, 2, 3, 4, 5, 6, 6, 6]
[0, 1, 2, 3, 4, 5, 6, 6, 6]
*****
La subsecuencia común más larga es: PRINTI
Su longitud es: 6
*****
Soluciones óptimas:
*****
PRINTI
PRINTI
PRINTI
PRINTI
PRINTI
PRINTI
PS C:\Users\darwi\OneDrive\Escritorio\Complejidad> 
```

## 3. Solución

La solución de la LCS entre

Cadena 1: springtime

Cadena 2: printing

Es: PRINTI