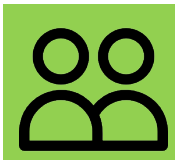
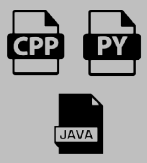
Taller en Sala Nro. 9
Programación Dinámica

En la vida real, la distancia de Levenshtein se utiliza para algoritmos de reconocimiento óptico de caracteres, es decir, pasar de imagen a texto. También se utiliza en correctores de ortografía, como el que trae Microsoft Word y en especial el de los teclados de los celulares, al igual que en procesamiento del lenguaje natural como Siri de Apple.

 **Trabajo en Parejas**	 **Hoy, plazo máximo de entrega**	 **Docente entrega código suelto en GitHub**
 **Sí .cpp, .py o .java**	 **No .zip, .txt, html o .doc**	 **Alumnos entregan código suelto por GitHub**

Ejercicio a resolver

1. Dadas dos cadenas de caracteres *a* y *b*, determine la distancia *Levenshtein* que hay entre ellas, es decir, la cantidad mínima de operaciones (insertar, remover o cambiar una letra) que se necesitan para transformar una en la otra utilizando programación dinámica.

```
public static int levenshtein(String a, String b) {
   // complete...
}
```

DOCENTE MAURICIO TORO BERMÚDEZ
Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627
Correo: mtorobe@eafit.edu.co

2. [Opcional] El problema de la **subsecuencia común más larga** es el siguiente. Dadas dos secuencias, encontrar la **longitud** de la secuencia más larga presente en ambas. Una subsecuencia es una secuencia que aparece en el mismo orden relativo, pero no necesariamente de forma contigua

Teniendo en cuenta lo anterior, realicen un programa que calcule la longitud de la subsecuencia común más larga a dos cadenas de caracteres

Ayudas para el Ejercicio 1 y 2……. Pág. 3

Ayudas para resolver el Ejercicio 1

Como un ejemplo, si tenemos las palabras "*carro*" y "*casa*" la distancia Levenshtein que hay entre ellas es 3:

1. Remover una letra: "carr"
2. Cambiar una letra: "casr"
3. Cambiar una letra: "casa"

Pista 1: Para observar la respuesta que entrega el algoritmo, utilice el simulador que se encuentra en http://www.let.rug.nl/kleiweg/lev/ . Configure `indel=1`, substitution=`1` y `swap=1`

Pista 2: Nótese que las operaciones y su orden pueden ser diferentes, pero lo importante es que el número mínimo de operaciones para transformar "carro" en "casa" son 3.

Pista 3: Asuman que las cadenas dadas están ambas completamente en minúscula o mayúscula.

Pista 4: Solucionen el siguiente problema para tener una mayor seguridad de que su implementación es correcta: http://www.spoj.com/problems/EDIST/

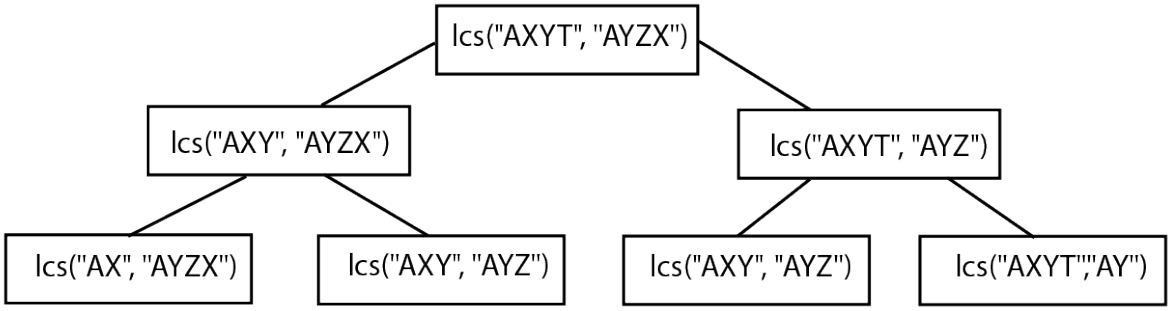
Ayudas para resolver el Ejercicio 2

Como un ejemplo, "abc", "abg", "bdf", "aeg" y "acefg" son subsecuencias de "abcdefg". Entonces, para una cadena de longitud n existen 2^n posibles subsecuencias.

DOCENTE MAURICIO TORO BERMÚDEZ
Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627
Correo: mtorobe@eafit.edu.co

Como un ejemplo, considere los siguientes ejemplos para el problema:

Para “ABCDGH” y “AEDFHR” es “ADH” y su longitud es 3. Para “AGGTAB” y “GXTXAYB” es “GTAB” y su longitud es 4. Una forma de resolver este problema es usando backtracking, como un ejemplo, para las cadenas “AXYT” y “AYZX”, dada una función recursiva lcs que resuelve el problema, se obtendría el siguiente árbol (parcial) de recursión:



Pista 1: Usando backtracking para ese problema, el problema lcs(“AXY”, “AYZ”) se resuelve dos veces. Si dibujamos el árbol de recursión completo, veremos que aparecen más y más problemas repetidos, así como en el caso de serie de Fibonacci. Este problema se puede solucionar guardando la soluciones, que ya se han calculado para los subproblemas, en una tabla; es decir, usando programación dinámica.

Pista 2: Complete el siguiente método:

```
// Precondición: Ambas cadenas x, y son no vacías
public static int lcsdyn(String x, String y) {
   …
}
```

DOCENTE MAURICIO TORO BERMÚDEZ
Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627
Correo: mtorobe@eafit.edu.co

DOCENTE MAURICIO TORO BERMÚDEZ
Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627
Correo: mtorobe@eafit.edu.co

¿Alguna inquietud?

CONTACTO

Docente Mauricio Toro Bermúdez
Teléfono: (+57) (4) 261 95 00 **Ext.** 9473
Correo: mtorobe@eafit.edu.co
Oficina: 19- 627

Agende una cita con él a través de **http://bit.ly/2gzVg10** , en la pestaña *Semana. Si no da clic en esta pestaña, parecerá que toda la agenda estará ocupada.*