



# Tecnológico de Monterrey

Instituto Tecnológico y de Estudios Superiores de Monterrey

## **Análisis y diseño de algoritmos avanzados (Gpo 601)**

### **E1. Actividad Integradora 1**

Nicole Dávila Hernández – A01784217

Campus Santa Fe

Víctor Manuel de la Cueva Hernández

28 de octubre 2024

# E1. Actividad Integradora 1

Al crear la función *substringComunMasLargo*, se utilizó una técnica de programación dinámica, ya que permite manejar comparaciones entre dos cadenas de manera eficiente. La idea de ocupar este método fue construir una matriz bidimensional donde cada celda representa la longitud de coincidencia acumulada entre caracteres específicos de las dos cadenas. Al iterar sobre cada carácter de ambas cadenas, en cada coincidencia incremento el valor en la celda correspondiente de la matriz, con base al valor diagonal anterior. Esto representa la longitud de la coincidencia acumulada hasta ese punto. Cuando encontramos una coincidencia más larga, almacenamos la posición final y actualizamos la longitud máxima de coincidencia, lo cual nos da la posición y longitud del *substring* común más largo.

La complejidad de este algoritmo es  $O(n * m)$ , donde  $n$  y  $m$  son las longitudes de las dos cadenas. Esta complejidad surge debido a que, en el peor de los casos, se necesita iterar sobre todos los pares posibles de caracteres entre ambas cadenas. Aunque este enfoque puede ser costoso en términos de memoria y tiempo para cadenas de gran tamaño, es una solución confiable y clara, adecuada para encontrar el *substring* común más largo.

Además, desarrollé la función *leeArchivo*, que permite leer archivos de texto y almacenar su contenido en una cadena. Este paso de preprocesamiento es fundamental para asegurar que los datos se lean de forma eficiente, especialmente si el archivo es largo. Esta función tiene una complejidad  $O(n)$ , donde  $n$  es la longitud del archivo en caracteres, ya que itera sobre cada línea del archivo para construir una sola cadena de texto completa.