

**Universidad del Valle de Guatemala**  
**Facultad de ingeniería**



**Corto 3: Generando "to be or not to be" en Python**

**Estefanía Elvira 20725**

**Guatemala 5 de Octubre del 2023**

### **Número de generaciones requeridas para converger en la frase objetivo**

- En nuestro experimento inicial, el algoritmo genético logró converger hacia la frase objetivo "to be or not to be" en tan solo 18 generaciones. Esto demuestra la eficacia del algoritmo para evolucionar rápidamente hacia soluciones óptimas o cercanas a óptimas, especialmente cuando las funciones de selección, reproducción y mutación están adecuadamente diseñadas.

### **Características de las funciones:**

- **Selección:** La función de selección evalúa a cada individuo de la población en función de su similitud con la frase objetivo. Aquí, la "aptitud" se mide por la cantidad de caracteres que coinciden con la frase objetivo en la misma posición. Posteriormente, se selecciona la mitad superior de individuos basándonos en su aptitud para la siguiente etapa. Esta técnica asegura que las secuencias más prometedoras se transmitan a la siguiente generación.
- **Reproducción:** Esta función toma dos individuos seleccionados al azar de la población y los combina para crear un hijo. Esta combinación se realiza eligiendo un punto de corte aleatorio y tomando la primera parte de un padre y la segunda parte del otro. Este método promueve la diversidad genética y permite que las características beneficiosas de ambos padres se combinen.
- **Mutación:** La función de mutación introduce cambios aleatorios en los genes de los individuos con una probabilidad definida por la tasa de mutación. Estos cambios aleatorios son esenciales para mantener la diversidad en la población y evitar el estancamiento. La mutación puede permitir que la población escape de óptimos locales y explore nuevas soluciones.

### **Preguntas a responder:**

#### **- ¿Qué sucede si el tamaño máximo de la población es demasiado bajo?**

Un tamaño de población muy bajo puede limitar la diversidad genética, lo que a su vez puede hacer que el algoritmo se estanque en óptimos locales y no encuentre la solución óptima. Además, con menos individuos, hay menos oportunidades de cruzar y combinar características beneficiosas.

#### **- ¿Qué ocurre si el tamaño máximo de la población es demasiado alto?**

Si bien una población más grande puede ofrecer una mayor diversidad genética, también puede aumentar significativamente el tiempo computacional y los recursos necesarios para cada generación. Además, tener una población excesivamente grande no garantiza necesariamente una convergencia más rápida o mejores soluciones.

**- ¿Qué sucede si la tasa de mutación es del 0%?**

Si la tasa de mutación es 0%, significa que no se introducirán cambios aleatorios en la población. Aunque esto puede parecer beneficioso para preservar soluciones prometedoras, en realidad puede llevar al estancamiento. La mutación es esencial para introducir nueva diversidad y permitir que el algoritmo explore soluciones fuera del espacio actual.

**- ¿Qué pasa si la tasa de mutación es del 50%?**

Una tasa de mutación tan alta puede introducir demasiado ruido y aleatoriedad en el proceso. Si bien esto podría aumentar la diversidad, también puede hacer que el algoritmo pierda rápidamente soluciones prometedoras y dificultar la convergencia hacia una solución óptima.

**- ¿Qué ocurre si la tasa de mutación es del 100%?**

Esto significaría que cada gen de cada individuo se mutaría en cada generación. Tal escenario haría que el algoritmo se comportara casi aleatoriamente, ya que todas las buenas características o genes serían reemplazados en cada generación. Es muy probable que el algoritmo no converja o tarde mucho tiempo en hacerlo.

**Link del repositorio de github:**

- <https://github.com/Estef072/Corto3-Modelacion.git>