

**Instituto Técnico Superior Córdoba**

**Tecnicatura Superior en Desarrollo de Software**

**Programación II**

**Prof.: Luis Romano**

**Estudiante: Fernando Gastón Gigena Parola**

## **Parcial 2: Complejidad algorítmica – Estructuras lineales**

- **Defina Big O y explique por qué es útil en el análisis de algoritmos.**

La notación Big O es una forma de medir cuanto tarda un algoritmo en ejecutarse dependiendo de qué tan grande sea el problema. No se fija en los detalles pequeños (como constantes o pasos extra), sino en la parte que más influye cuando el problema crece mucho.

Es útil porque sirve para comparar algoritmos de manera general, sin importar la computadora o el lenguaje de programación utilizado, de esta forma podemos saber cuál es más eficiente cuando el problema se hace grande y elegir que conviene usar.

- **Describa la diferencia entre una pila (LIFO) y una cola (FIFO). Implemente dichas estructuras en python, explicando cada una de sus operaciones.**

**LIFO: último en entrar, primero en salir.** Éste brinda un ordenamiento basado en el tiempo de permanencia en la colección. Los ítems más nuevos están cerca al tope y los más viejos están más cerca de la base.

**FIFO: (first-in first-out),** también conocido como el primero en llegar es el primero en ser atendido.

Link a los archivos en github: <https://github.com/GastonGigena/Complejidad-algor-tmica-Estructuras-lineales.git>