

# CS2013: Programación III

## Laboratorio: Complejidad Algorítmica I

### Laboratorio 7B

José Chávez

# Agenda

- Complejidad Algorítmica
- Notación Big-O

# Ejercicios

## Ejercicio 1:

Suponga que tiene la secuencia **ordenada** de números enteros A.  
Implemente un algoritmo (**sin utilizar memoria adicional**) que verifique si existe un par de números los cuales suman 10.

Por ejemplo:

- 1,3,4,5,8 (Falso)
- 2,3,4,5,5 (Verdadero)
- -3,1,1,2,3,5,6,10,13 (Verdadero)

No utilizar ChatGPT, ni GitHub Copilot, etc.

## Ejercicio 2:

Suponga que tiene la secuencia de números enteros A. Implemente un algoritmo que verifique si existe un par de números los cuales suman 10.

Por ejemplo:

- 1,3,4,5,8 (Falso)
- 2,3,4,5,5 (Verdadero)
- -3,1,1,2,3,5,6,10,13 (Verdadero)
- 9,1,2,3,5,1,0 (Verdadero)

No utilizar ChatGPT, ni GitHub Copilot, etc.

## Ejercicio 3:

Suponga que tiene la secuencia de números enteros A. Implemente un algoritmo que detecte el primer bucle y que retorne el elemento recurrente.

Por ejemplo:

- 1,3,2,4,1,5,3,2 (1)
- 5,1,2,4,5 (5)
- 1,2,3,4,5,8 (-1)

No utilizar ChatGPT, ni GitHub Copilot, etc.

## Ejercicio 4:

Comparar los tiempos de ejecución para los algoritmos *Insertion Sort* y *Bubble Sort*, e indicar cuál es el más eficiente para ordenar. Comprueba para:

- El peor de los casos (la secuencia está en orden inverso)
- El mejor de los casos (la secuencia ya está ordenada)

# Resumen

En esta sesión se trataron los tópicos siguientes:

- Notación Big-O
- Algoritmos de tiempo constante, lineal, cuadrático y logarítmico.

