

# Ayudantía 8

---

ESTRUCTURA DE DATOS Y ALGORITMOS

PROFESOR: YERKO ORTIZ

AYUDANTE: VICENTE DÍAZ

SECCIÓN 6

# Contacto

---

Discord: trapitokid

Wsp: +569 5749 6014

LoL: Nozomi

BarroLuco#woof



# Problema 1

---

```
class Estudiante {  
    String nombre;  
    double nota;  
  
    Estudiante(String nombre, double nota) {  
        this.nombre = nombre;  
        this.nota = nota;  
    }  
  
    public String toString() {  
        return nombre + " (" + nota + ")";  
    }  
}
```



Dado un arreglo de objetos Estudiante con nombre y nota (por ejemplo, ("Benjamin", 7.0), ("Catalina", 6.0)), implementa tres algoritmos de ordenamiento (Bubble Sort, Selection Sort, Insertion Sort) para ordenar por nota.

- 1) Implementa los tres algoritmos.
- 2) Muestra el orden antes y después del ordenamiento.
- 3) ¿Cuál es estable? ¿Cómo lo sabes?

# Problema 2

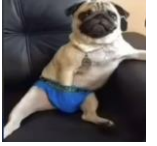
---

```
for (int i = 0; i < n-1; i++) {  
    int min = i;  
    for (int j = i+1; j < n; j++) {  
        if (arr[j] < arr[min]) {  
            min = j;  
        }  
    }  
    if (min != i) {  
        int temp = arr[i];  
        arr[i] = arr[min];  
        arr[min] = temp;  
    }  
}
```



Analiza fragmentos de código de algoritmos y deduce si son estables, sin ejecutarlos.

- 1) ¿Qué algoritmo es?
- 2) ¿Es estable? ¿Por qué?
- 3) ¿Cómo lo harías estable?



# Problema 3

---

Implementa una versión modificada de Selection Sort que sea estable. Luego responde las siguientes preguntas:

- 1) ¿Qué implica que un algoritmo de ordenamiento sea estable?
- 2) ¿Cuándo importa que sea estable?
- 3) ¿Cuántos swaps hace Bubble Sort en el peor caso?
- 4) ¿Qué relación hay entre estabilidad y número de swaps?

**Hint:** En vez de hacer swap, mueve el mínimo elemento insertándolo y desplazando los demás.