

# Algoritmos de Ordenación

## Ejercicio 1

---

Crea un menú con las siguientes opciones para trabajar con un array que puede almacenar máximo de 10 números:

- Introducir valor: se pide al usuario un número, y si ese número no existe en el array, se introducirá en el primer espacio libre disponible. Si no hay espacios disponibles, se dará un aviso de error.
- Buscar valor: se pide un número, y si existe en el array nos dice en qué posición está. Si no, da un mensaje de error.
- Borrar valor: se pide un número, si está en el array lo elimina. Si no, da un aviso de error.
- Salir.

## Ejercicio 2

---

Lo mismo que antes, pero el array tiene que estar SIEMPRE ordenado. Para ello, cada vez que se hace una inserción o un borrado, se llamará a un método bubblesort () que reordenará de mayor a menor el array.

## Ejercicio 3

---

Lo mismo que antes, pero se tienen que implementar Búsquedas Binarias.

## Ejercicio 4

---

Tienda de mascotas. Cada mascota tiene una id, un tipo de animal, un color y si está vacunada o no. Quiero poder buscar, borrar e insertar mascotas. Guarda las mascotas en un Array, pero manténlo siempre ordenado por bubblesort (). Usa la id como criterio de ordenación. Todas las búsquedas tienen que ser binarias.