



# Fundamentos de programación

## 0.8 Estructuras de datos y algoritmos

### 0.8.5 Algoritmos de búsqueda

#### Importación de Librerías

- **Librería utilizada:** `crypto`
- **Propósito:** Obtener valores aleatorios enteros
- **Uso adicional:** Firmar o cifrar mensajes o datos (aunque no se utiliza en este contexto).

#### Búsqueda Secuencial

- **Descripción:** Recorrido de un array desordenado elemento por elemento hasta encontrar el valor buscado.
- **Implementación:**
  - a. Crear un array con 10 millones de elementos aleatorios.
  - b. Utilizar un bucle **while** para recorrer el array.
  - c. Comparar cada elemento con el valor buscado.
  - d. Salir del bucle si el elemento es encontrado.
- **Complejidad:**  $O(n)$
- **Ejemplo de Código:**

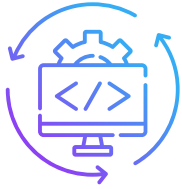
```
let encontrado = false;
let index = 0;
while (!encontrado && index < lista.length) {
  if (lista[index] === busqueda) {
    encontrado = true;
  }
  index++;
}
```

#### Búsqueda Binaria

- **Descripción:** Algoritmo que requiere un array ordenado y divide el array en mitades para encontrar el valor buscado.
- **Implementación:**
  - a. Ordenar el array.
  - b. Dividir el array en mitades repetidamente hasta encontrar el valor.
- **Requisito:** El array debe estar ordenado.
- **Complejidad:**  $O(\log n)$
- **Ejemplo de Código:**

```
let inicio = 0;
let fin = lista.length - 1;
let encontrado = false;

while (inicio <= fin && !encontrado) {
  let mitad = Math.floor((inicio + fin) / 2);
  if (lista[mitad] === busqueda) {
    encontrado = true;
  } else if (lista[mitad] < busqueda) {
    inicio = mitad + 1;
  } else {
    fin = mitad - 1;
  }
}
```



## 0.8.5 Algoritmos de búsqueda

### Benchmarks

- Herramienta: `console.time` y `console.timeEnd`
- Propósito: Medir el tiempo de ejecución de los algoritmos
- Ejemplo de Uso:

```
console.time('BusquedaSecuencial');  
// Código de búsqueda secuencial  
console.timeEnd('BusquedaSecuencial');
```

```
console.time('BusquedaBinaria');  
// Código de búsqueda binaria  
console.timeEnd('BusquedaBinaria');
```