

Algoritmos de Búsqueda en Java

Buscar se trata de encontrar datos de manera eficiente. Exploremos la búsqueda binaria y la búsqueda lineal.

1. Búsqueda Binaria: Eficiencia Logarítmica

La búsqueda binaria funciona con arreglos ordenados, reduciendo el espacio de búsqueda a la mitad en cada paso, con un tiempo de $O(\log n)$.

Implementación en Java

```
public class BinarySearch {  
    public static int binarySearch(int[] arr, int target) {  
        int left = 0, right = arr.length - 1;  
        while (left <= right) {  
            int mid = left + (right - left) / 2;  
            if (arr[mid] == target) return mid;  
            if (arr[mid] < target) left = mid + 1;  
            else right = mid - 1;  
        }  
        return -1;  
    }  
  
    public static void main(String[] args) {  
        int[] arr = {2, 3, 4, 10, 40, 50};  
        int target = 10;  
        System.out.println("Encontrado en: " + binarySearch(arr, target));  
    }  
}
```

Salida: Encontrado en: 3

2. Búsqueda Lineal: Línea Base Simple

La búsqueda lineal verifica cada elemento secuencialmente, con un tiempo de $O(n)$ —útil para datos no ordenados.

Implementación en Java

```
public class LinearSearch {  
    public static int linearSearch(int[] arr, int target) {  
        for (int i = 0; i < arr.length; i++) {  
            if (arr[i] == target) return i;  
        }  
        return -1;  
    }  
  
    public static void main(String[] args) {  
        int[] arr = {10, 20, 80, 30, 60, 50, 110};  
        int target = 30;  
        System.out.println("Encontrado en: " + linearSearch(arr, target));  
    }  
}
```

Salida: Encontrado en: 3