

Algoritmo Quicksort

JORGE GABRIEL LEYVA DURÁN

UNIVERSIDAD AUTÓNOMA
METROPOLITANA

UNIDAD IZTAPALAPA

VIERNES 28 DE MARZO DE 2025.

Contenido

PRINCIPIO DE “DIVIDE Y VENCERÁS”

ALGORITMO QUICKSORT

EJEMPLO

VENTAJAS Y DESVENTAJAS

IMPLEMENTACIÓN EN C

Principio de “divide y vencerás”

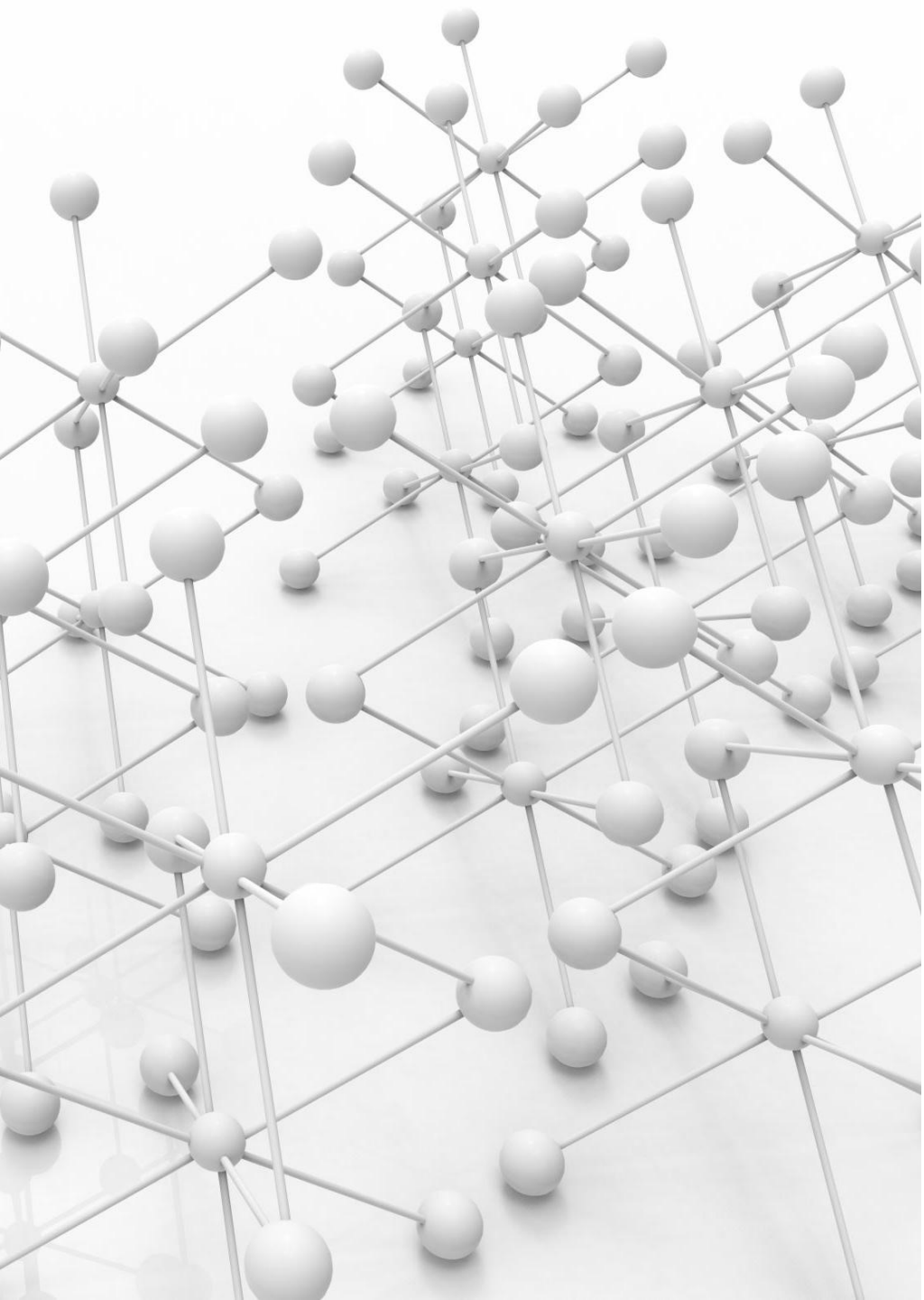


- Es un método basado en la resolución recursiva de un problema, haciéndole subdivisiones más sencillas.
- Dividir un problema difícil en varios problemas sencillos.
- Es la base de los algoritmos de ordenamiento tales como: Mergesort, Quicksort, etc.

Algoritmo Quicksort

DESARROLLADO POR
ANTONY R. HOARE EN 1959





Algoritmo Quicksort

1. **Seleccionar un pivote:** Elegir un elemento del arreglo como pivote. Generalmente, se utiliza el primer o el último elemento del arreglo para simplificar el proceso.
2. **Particionar el arreglo:**
 - i. Recorrer el arreglo y mover todos los elementos **menores al pivote** hacia la izquierda.
 - ii. Recorrer el arreglo y mover todos los elementos **mayores o iguales al pivote** hacia la derecha.
3. **Colocar el pivote en su posición final:** Una vez que los elementos han sido separados, colocar el pivote en su posición correcta dentro del arreglo, de modo que los elementos a la izquierda sean menores y los de la derecha mayores.
4. **Aplicar recursividad:** Repetir los pasos 2 y 3 de forma recursiva en los subarreglos izquierdo y derecho hasta que cada subarreglo contenga un solo elemento o ninguno.
5. **Finalizar:** El arreglo estará completamente ordenado cuando no queden más particiones por realizar.

Ejemplo

- Supongamos que tenemos el siguiente arreglo: [42, 124, 23, 5, 89, -1, 44, 643, 34]
- Seleccionamos como pivote: 42 (primer elemento)

42 124, 23, 5, 89, -1, 44, 643, 34



23, 5, -1, 34, 42 124, 89, 44, 643

23, 5, -1, 34, 42, 124, 89, 44, 643

23 5, -1, 34

42

124 89, 44, 643

5, -1, 23 34

42

89 44, 124 643

-1 5 23 34

42

44, 89 124 643

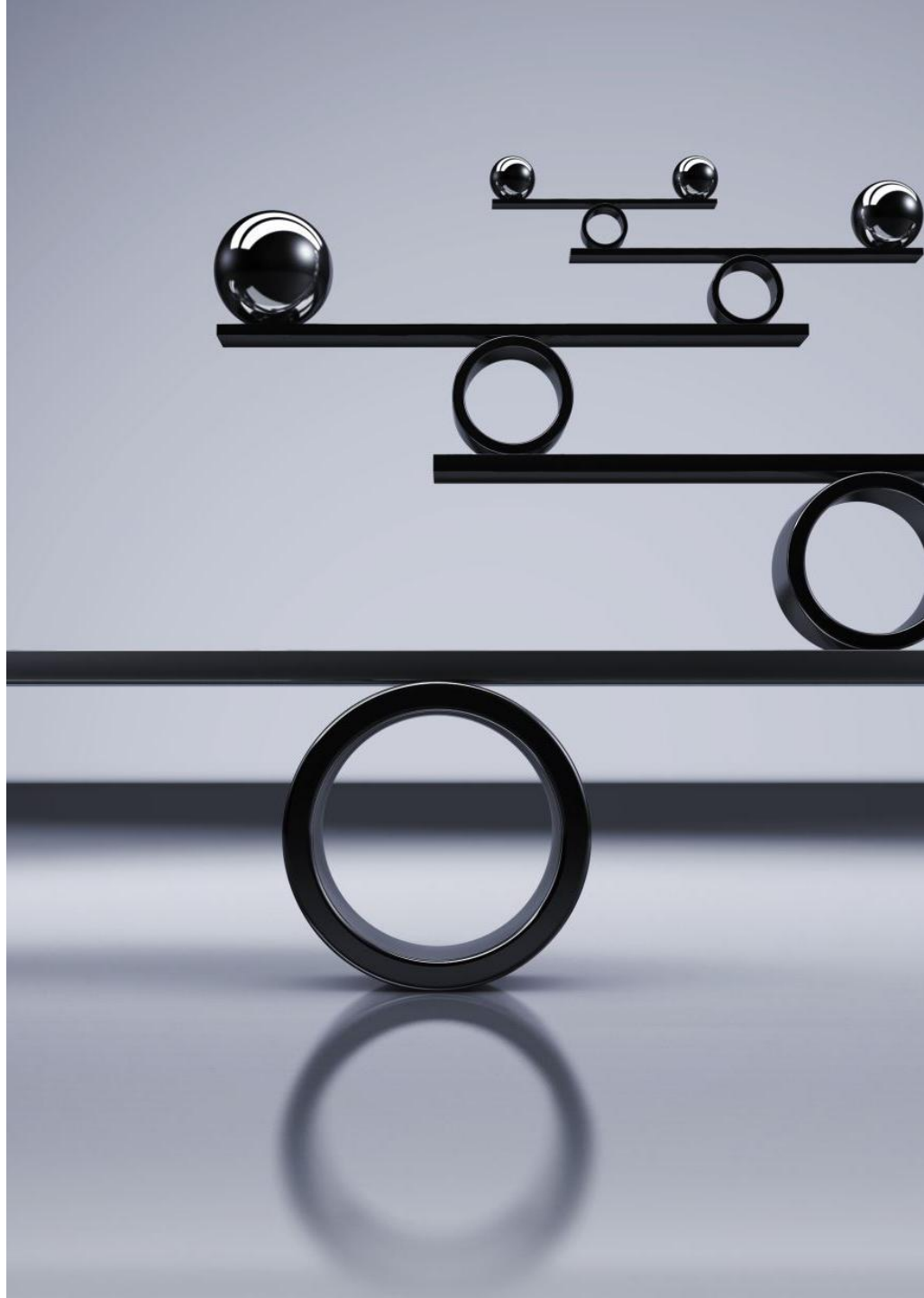
Ventajas y Desventajas

- Ventajas:

1. Por lo general, la complejidad es de $O = (n \log n)$
2. Uso eficiente de memoria.
3. Fácil implementación.
4. Divide y vencerás.

- Desventajas:

1. La complejidad en el peor de los casos es de $O = n^2$.
2. Dependencia del pivote, y por tanto la complejidad puede variar.
3. Uso intensivo de la recursión.





Implementación en C