



UNIVERSIDAD AUTÓNOMA DE CAMPECHE.

ALUMNOS: GABRIEL ALEJANDRO MOO MARTÍNEZ.

CARLOS ALBERTO DOMÍNGUEZ BORGES.

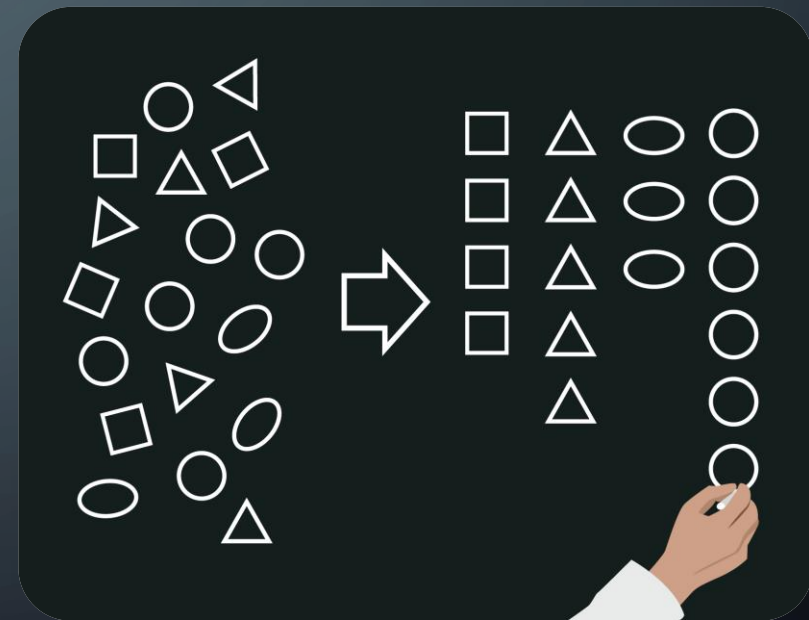
RAFAEL GILBERTO HERNÁNDEZ JIMÉNEZ.

DOCENTE: JOSÉ ÁNGEL GARCÍA REYES.

MÉTODO DE ORDENAMIENTO QUICKSORT.

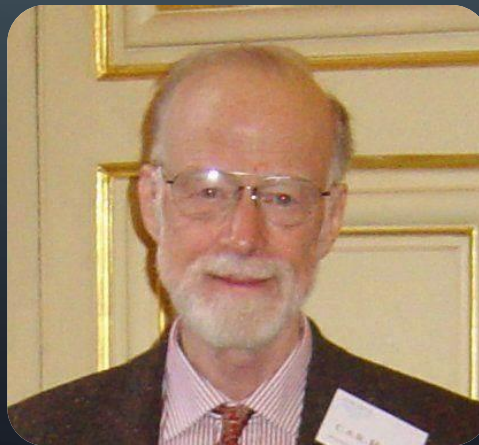
MATERIA: ESTRUCTURA DE DATOS II.

GRADO Y GRUPO: 5° "A".



HISTORIA.

- El algoritmo Quicksort tiene una historia interesante y es uno de los métodos de ordenamiento más utilizados en la programación.
- El Quicksort, también conocido como ordenamiento rápido, fue creado por el científico británico en computación Charles Anthony Richard Hoare en 1960. Este algoritmo sigue el principio de “divide y vencerás”, lo que significa que divide el conjunto de elementos a ordenar en subconjuntos más pequeños, los ordena por separado y luego los combina para obtener la lista ordenada.



FUNCIONAMIENTO.

1. Elección del pivote: El primer paso es elegir un elemento de la matriz como pivote. El pivote es crucial para el rendimiento del algoritmo. Generalmente, se selecciona el primer elemento de la matriz, pero también se pueden usar otras estrategias para elegir el pivote.
2. Partición: A continuación, se reorganizan los demás elementos de la matriz de manera que los elementos más pequeños que el pivote estén a su izquierda y los elementos más grandes estén a su derecha. Esto se llama “partición”.
3. Recursión: Después de la partición, tenemos dos subconjuntos: uno con elementos más pequeños que el pivote y otro con elementos más grandes. Luego, aplicamos Quicksort de forma recursiva a cada uno de estos subconjuntos.
4. Combinación: Finalmente, cuando todos los subconjuntos están ordenados, combinamos los resultados para obtener la matriz completa ordenada.

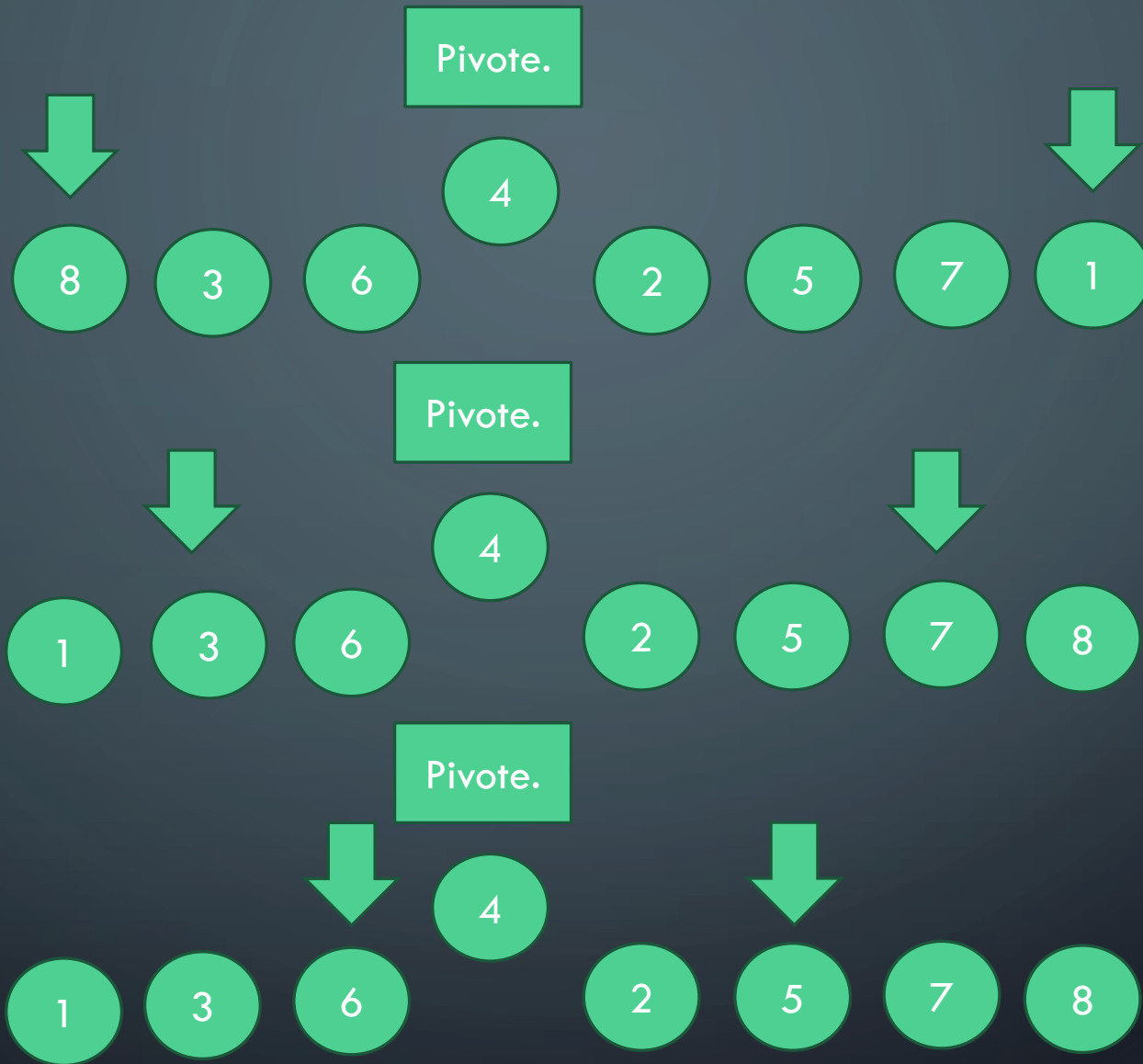
VENTAJAS.

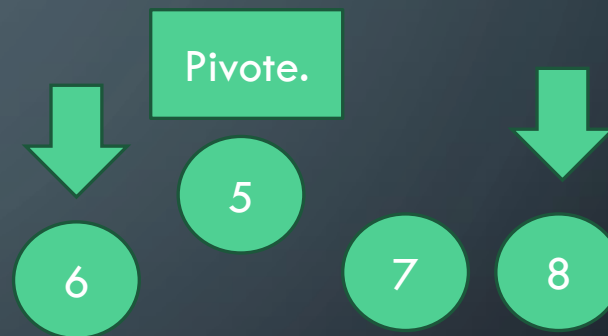
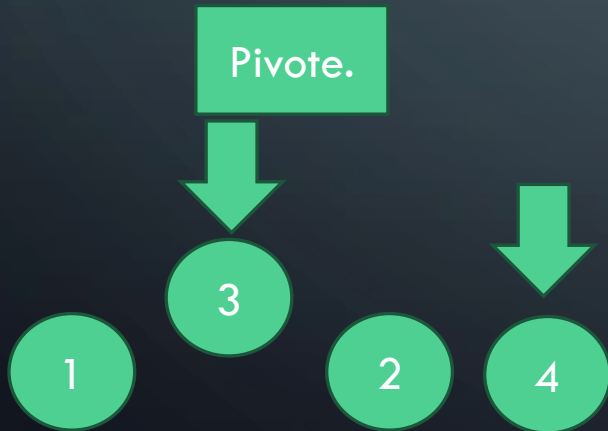
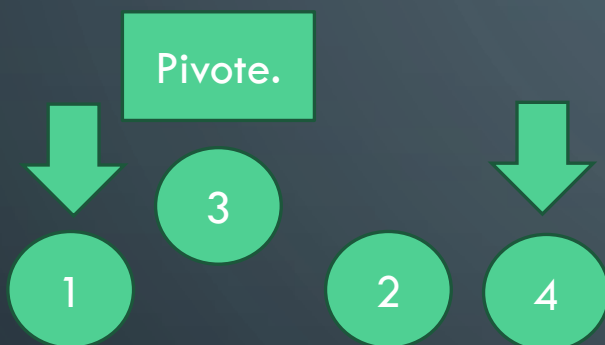
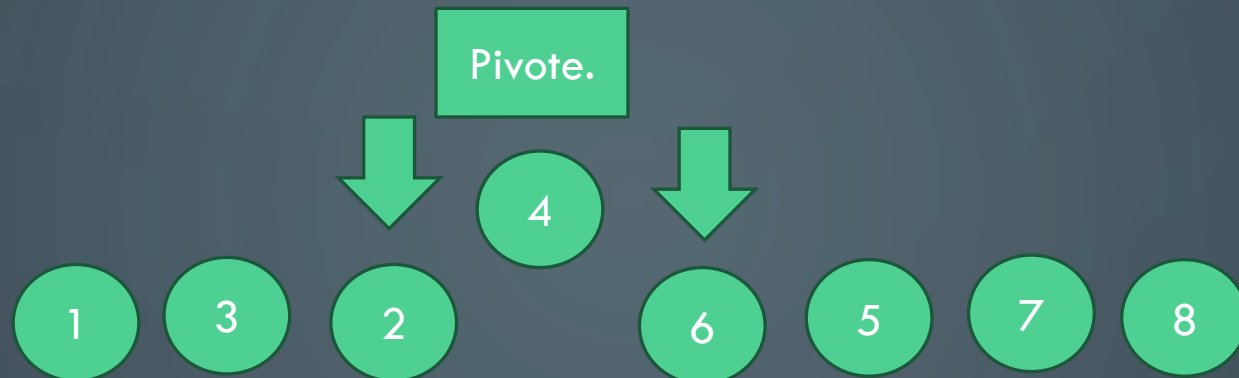
- Eficiencia promedio: En promedio, Quicksort tiene un rendimiento muy bueno. Su complejidad de tiempo es $O(n \log n)$ en el caso promedio, lo que lo hace eficiente para ordenar grandes conjuntos de datos.
- Divide y conquista: Quicksort utiliza una estrategia de “dividir y conquistar”, lo que significa que divide la lista en subconjuntos más pequeños y luego los combina. Esto maximiza la eficiencia y minimiza el tiempo de ejecución.
- In-place: Quicksort ordena los elementos directamente en su ubicación original, sin requerir memoria adicional para copias temporales. Esto es útil cuando la memoria es escasa.

DESVENTAJAS.

- Implementación complicada: Aunque es eficiente, la implementación de Quicksort puede ser un poco más complicada en comparación con otros algoritmos de ordenamiento, como el Bubble Sort o el Insertion Sort.
- Recursividad intensiva: Quicksort utiliza recursión, lo que puede consumir muchos recursos en pilas de llamadas. En casos extremos, esto podría llevar a un desbordamiento de pila (stack overflow).
- Variabilidad en el rendimiento: La diferencia entre el mejor y el peor caso puede ser significativa. Si el pivote elegido no divide bien la lista, el rendimiento puede degradarse a $O(n^2)$ en el peor caso.

EJEMPLO.





Pivote.



Pivote.



PREGUNTAS.

- 1.-¿Quién creo el método de ordenamiento Quicksort?
- 2.-¿En que año fue creado?
- 3.-¿Qué principio sigue el método de ordenamiento Quicksort?
- 4.-¿En que consiste este principio?
- 5.-¿Cuál es el primer paso del funcionamiento?
- 6.-¿En que consiste el paso de partición?
- 7.-¿En que consiste el paso de recursión?
- 8.-¿En que consiste el paso de combinación?
- 9.-Menciona dos ventajas del método de ordenamiento Quicksort.
- 10.-Menciona dos desventajas del método de ordenamiento Quicksort.