

Algoritmo de ordenamiento **Introsort**



Integrantes:

- Mauro Chumpitaz
- Eduardo Medina
- Sebastian Escalante

Profesor:

- Luis Talavera Ríos

1

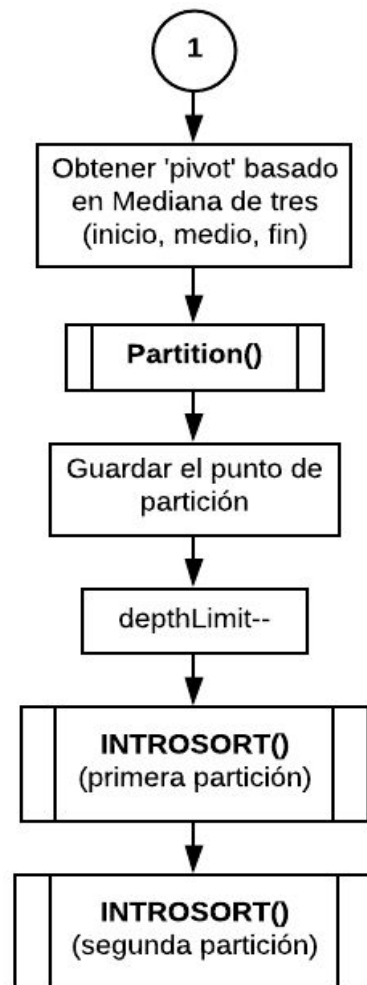
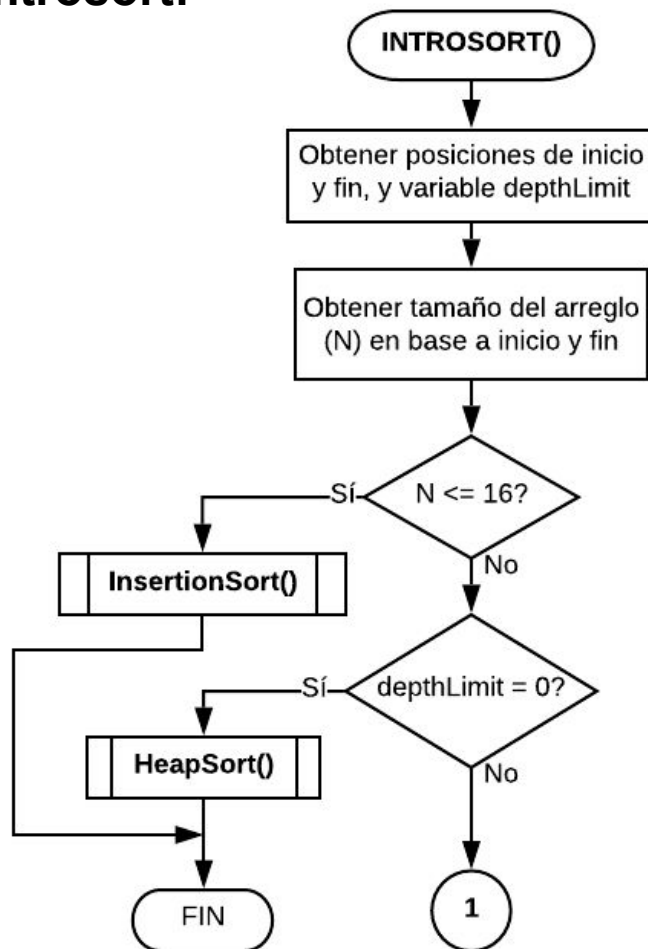
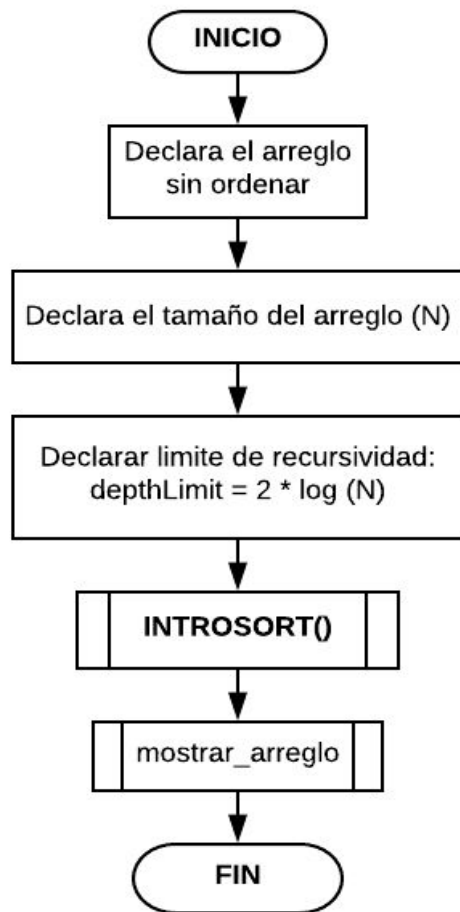
¿Cómo funciona?

Definición y explicación

Introsort es un algoritmo de ordenamiento híbrido **basado en Quicksort, Heapsort e Insertion sort**, y es utilizado por la librería estándar de C++.

“

Proceso del algoritmo Introsort:





Comparaciones

¿Por qué Heapsort y no Merge sort?

Por requerimientos de espacio de memoria. Mergesort emplea $O(N)$, mientras Heapsort es $O(1)$ por trabajar 'in-place'.

¿ Por qué Insertion sort y no Bubble*?

Por el hecho que es el mejor algoritmo de ordenamiento para arreglos pequeños y por su adaptabilidad de ordenar subarreglos parcialmente ordenados.

¿Por qué un arreglo de 16 elementos como punto de corte?

Éste valor fue obtenido en base a resultados experimentales de varias pruebas e investigaciones sobre Introsort.

2

Programarlo

Enlace del repositorio: <https://github.com/Marcuss17/IntroSort>

3

Complejidad y estabilidad

Recursos de espacio, tiempo y estabilidad.



Complejidad de espacio

Quicksort - $O(\log(n))$

Debido a la recursión, se necesita llamar a 'stacks' auxiliares para la implementación de estas iteraciones.

Heapsort - $O(1)$

Se implementa de manera 'in-place', es decir, a parte del espacio utilizado inicialmente para almacenar el arreglo, no usa memoria adicional a esa.

Insertion sort - $O(1)$

De la misma manera que Heapsort, Insertion sort no necesita de mayor espacio de memoria.

IntroSort - $O(\log(n))$

Al igual que el QuickSort puede necesitar de espacio auxiliar en el "stack" debido a la recursión.

Big-O

	Time Complexity (Best Case)	Time Complexity (Avg)	Time Complexity (Worst Case)
Quicksort	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$	$O(n^2)$
Heapsort	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$
Insertion sort	$O(n)$	$O(n^2)$	$O(n^2)$
IntroSort	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$



Estabilidad

IntroSort

No Estable - Al estar compuesto por algoritmos de ordenamiento no estables, como el HeapSort y Quicksort, entonces Introsort tampoco es estable.

4

Limitaciones

Cuando no se debería usar IntroSort.

Limitaciones



No usar este algoritmo:

- Cuando se requiera ordenar un arreglo de forma estable. Ya que este no lo es, y por lo tanto no respeta el orden original de los datos repetidos.
- Ejemplo: Una base de datos de compradores a los que se busca darle un premio según la hora en que llegaron, y se desea encontrar el nombre del ganador.



Bibliografía

- <https://www.geeksforgeeks.org/know-your-sorting-algorithm-set-2-introsort-cs-sorting-weapon/>
- Cormen, T., Leiserson, Ch., Rivest, R., Stein C.(1989). *Introduction to Algorithms*. MIT Press.



Gracias!

Preguntas:

1. *¿Para un arreglo de 7 elementos que tipo de sort se usaría?*
2. *¿ Usarían el Introsort para ordenar un unordered multimap?*