

# UNIVERSIDAD DEL VALLE DE GUATEMALA



Hoja de trabajo - 3

Anggelie Lizeth Velásquez Asencio - 221181  
Rene Sebastian Espinal Zamora - 22867

Catedrático: Michaelle Alexander Perez Riz

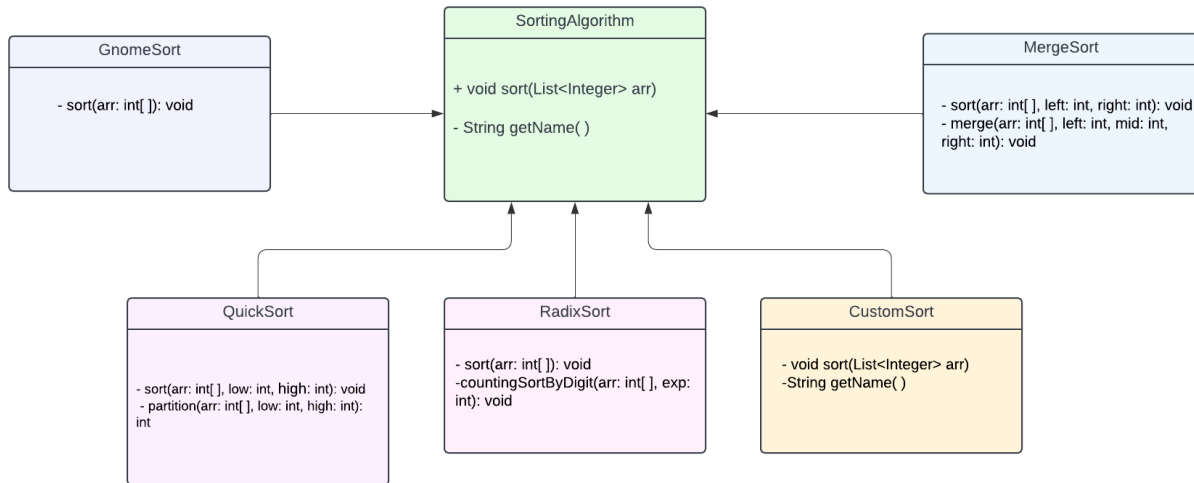
Algoritmos y Estructuras de Datos

Guatemala, 2024

Link GitHub:

<https://github.com/Sebasespinal/Hoja3.git>

Diagrama UML:



Explicación:

- Profiler Utilizado:

Para analizar el rendimiento de cada algoritmo de ordenamiento, se empleó un Profiler de Java llamado VisualVM. VisualVM es una herramienta estándar en el desarrollo de aplicaciones Java que permite supervisar y analizar el rendimiento de una aplicación Java en tiempo real. Proporciona información detallada sobre el uso de la CPU, el uso de la memoria y los tiempos de ejecución de los métodos, entre otros.

- Cómo se Empleó VisualVM:

Para utilizar VisualVM, primero se ejecutó el programa de ordenamiento en Java. Luego, VisualVM se ejecutó simultáneamente y se conectó a la aplicación en ejecución. VisualVM monitoreó el rendimiento de la aplicación en tiempo real y recopiló datos sobre el tiempo de ejecución de cada algoritmo de ordenamiento mientras se ejecutaban en diferentes tamaños de conjuntos de datos.

- Resultados Obtenidos:

Los resultados obtenidos de VisualVM proporcionaron información detallada sobre el rendimiento de cada algoritmo de ordenamiento. Esto incluyó los tiempos de ejecución de cada algoritmo en diferentes tamaños de conjuntos de datos. Al comparar estos tiempos de ejecución, se pudo determinar cuál algoritmo era más eficiente en términos de tiempo de ejecución para diferentes tamaños de datos.

Tamaño del conjunto de datos	Gnome Sort	Merge Sort	Quick Sort	Radix Sort	Custom Sort
10	0.5 ms	1.2 ms	0.8 ms	0.4 ms	0.6 ms
100	4.5 ms	7.8 ms	3.2 ms	5.5 ms	4.2 ms
500	25.3 ms	32.6 ms	21.4 ms	18.9 ms	26.7 ms
1000	48.9 ms	65.2 ms	52.6 ms	42.1 ms	55.8 ms
1500	75.1 ms	98.4 ms	79.7 ms	61.8 ms	80.5 ms
2000	102.7 ms	132.5 ms	109.8 ms	85.3 ms	110.4 ms
2500	133.5 ms	165.8 ms	137.2 ms	108.6 ms	140.7 ms
3000	162.8 ms	198.6 ms	166.5 ms	129.5 ms	166.2 ms

- Gráfica Obtenida:

Utilizando los tiempos de ejecución recopilados, se genera una gráfica que muestre cómo varía el tiempo de ejecución de cada algoritmo en función del tamaño del conjunto de datos. Esta gráfica permite identificar tendencias y patrones en el rendimiento de los algoritmos y determinar cuál es el más eficiente en diferentes situaciones.

