Universidad José Cecilio del Valle

Universidad José Cecilio Del Valle

**Proyecto I Parcial**

**Bubble Sort Vs Merge Sort**

Alumno: Alessandro Galindo – 2017110421

Ingeniero: Kenneth Vittetoe

Clase: Estructura de datos

Fecha: 16 de octubre del 2019

Índice

[Introducción 3](#_Toc22060903)

[Bubble Sort u ordenamiento de burbuja 3](#_Toc22060904)

[Código 4](#_Toc22060905)

[Merge Sort u ordenamiento por mezcla 4](#_Toc22060906)

[Código 5](#_Toc22060907)

[Comparación de Algoritmos 6](#_Toc22060908)

[Se implementó un Arreglo de 10 para hacer la comparación 6](#_Toc22060909)

[Se implementó un arreglo de 100 para hacer la comparación 6](#_Toc22060910)

[Se implementó un arreglo de 1000 para hacer la comparación 7](#_Toc22060911)

[Conclusiones 7](#_Toc22060912)

# Introducción

Los algoritmos de ordenamiento se asemejan al trabajo que hace un registro, si bien sabemos, los registros generales están conformados de forma ordenada ya sea alfabéticamente o numéricamente dependiendo de que se trate este mismo.

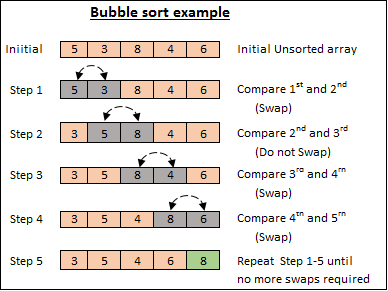
Estos Algoritmos nos ayudan o nos muestra un camino más fácil de hacer las cosas, estos son: El ordenamiento burbuja (Bubble Sort) y el ordenamiento de mezcla (Merge Sort).

En el siguiente informe se le estará detallando el funcionamiento de cada uno de estos métodos de ordenamiento con su respectiva definición y código en el lenguaje Java, así como también se le estará mostrando las diferentes pruebas realizadas con los códigos de cada uno y asi hacer la comparación de estos dos métodos de ordenamiento.

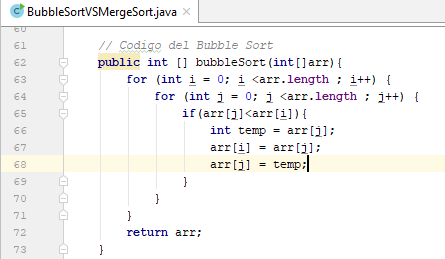
# Bubble Sort u ordenamiento de burbuja

Este algoritmo de burbuja consiste en estar haciendo una comparación de números ya sea desde el número mayor comenzando con la primera posición e ir número por numero hasta encontrar el número mayor, Si este es realmente el mayor de todo el vector se llevará hasta la última casilla, si no es así, será reemplazado por uno mayor que él. Este procedimiento se realizará hasta haber ordenado todas las casillas del vector.

Una de las deficiencias del algoritmo es que ya cuando ha ordenado parte del vector vuelve a compararlo cuando esto ya no es necesario. Dado un vector a1, a2, a3, ... an-1) Comparar a1 con a2 e intercambiarlos si a1>a2 (o a12) 2) Seguir hasta que todo se haya comparado an-1 con an3) Repetir el proceso anterior n-1 veces



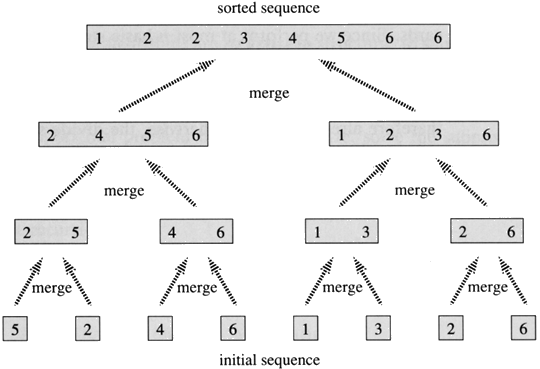
## Código



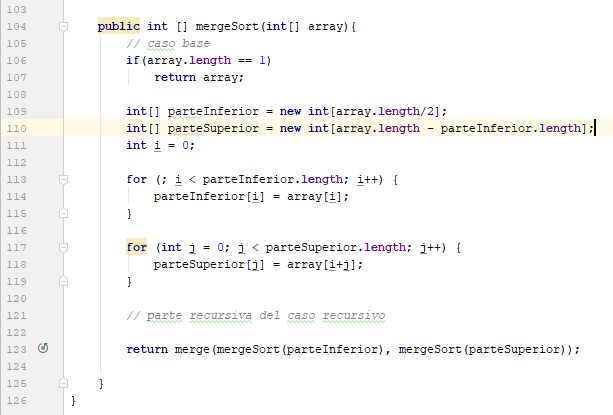
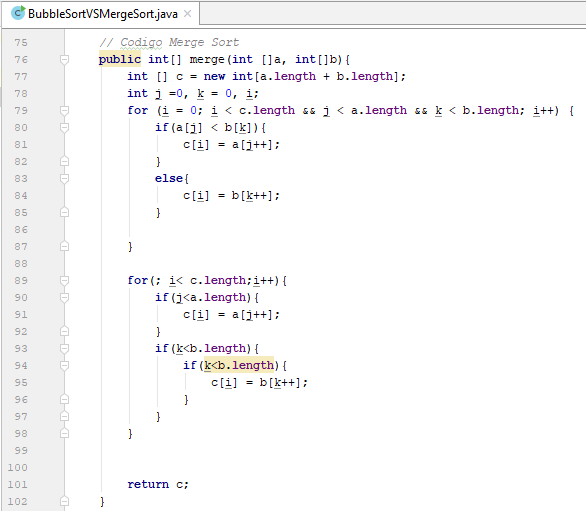
# Merge Sort u ordenamiento por mezcla

Es un algoritmo que utiliza la recursividad de un conjunto de elementos que se dividen entre dos. Se ordena cada parte de forma separada y al final se combinan los dos resultados en el arreglo original. También utiliza en principio “Divide y vencerás”.

Consiste en dividir el problema a resolver en subproblemas del mismo tipo que a su vez se dividirán, mientras no sean suficientemente pequeños o triviales.



## Código



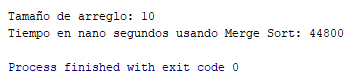
# Comparación de Algoritmos

## Se implementó un Arreglo de 10 para hacer la comparación

* **Bubble Sort**



* **Merge Sort**

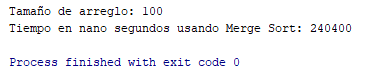


## Se implementó un arreglo de 100 para hacer la comparación

* **Bubble Sort**

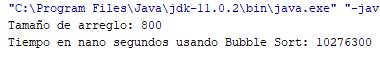


* **Merge Sort**

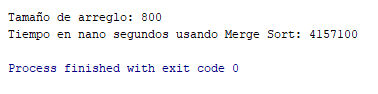


## Se implementó un arreglo de 800 para hacer la comparación

* **Bubble Sort**



* **Merge Sort**



# Grafico Comparativo

# Conclusiones

Como conclusión a este proyecto podemos ver que a pesar de que el método de ordenamiento Bubble Sort es el que tiene mas trayectoria en la computación no alcanzo a superar el tiempo que hizo el método de ordenamiento Merge Sort, este fue, de acuerdo a las pruebas realizadas el que supero por 2 veces al Bubble Sort en el cual queda comprobado que no solo porque el algoritmo de Bubble Sort sea más corto y simplificado significa que es el más eficiente al momento de la verdad.

Entonces podemos concluir y recomendar a los nuevos programadores que utilicen el ordenamiento de mezcla **Merge Sort** para hacer más eficiente su trabajo.

# Bibliografía

TutosPOO (Merge Sort) - Recuperado el 14 de octubre de 2019, de

<https://tutospoo.jimdo.com/tutoriales-java/m%C3%A9todos-de-ordenaci%C3%B3n/ordenaci%C3%B3n-por-combinaci%C3%B3n-mergesort/>

[devs4j](https://devs4j.com/) (Bubble Sort) - Recuperado el 14 de octubre de 2019, de

<https://devs4j.com/2018/12/19/algoritmo-de-la-burbuja-en-java-bubble-sort/>

EcuRed (Bubble Sort). Recuperado el 14 de octubre de 2019, de

https://www.ecured.cu/Ordenamiento\_de\_burbuja

EcuRed (Merge Sort) - Recuperado el 14 de octubre de 2019, de

https://www.ecured.cu/MergeSort

Codigo GitHub - https://github.com/Alessandro0919/ProyectoEstructura.git