Integrantes:

Daniel Alfonso Garcia Pilimur - da.garciap1@uniandes.edu.co - 202012183

Sebastian Andres Ospino Salinas - sa.ospino@uniandes.edu.co - 201913643

Rodrigo Romero Prada - r.romerop@uniandes.edu.co – 201923993

|  |  |  |
| --- | --- | --- |
| Fuente de datos | Arraylist [ms] | Singlelinkedlist [ms] |
| Películas (MoviesDetailsCleaned) | 44.863 | 45.946 |
| Elenco (MoviesCastingRaw) | 23.360 | 24.752 |

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arraylist [ms] | Singlelinkedlist [ms] |
| SelectionSort | 3264.078 | 313465.625 |
| InsertionSort | 3457.302 | 454585 |
| ShellSort | 65.123 | 6530.476 |

* ¿Qué diferencias se observan en desempeño de la carga de datos entre arreglo (*Arraylist*) y lista sencillamente encadenada (*Singlelinkedlist*)?: No hay mucha diferencia; la diferencia entre los dos tipos de lista en la carga de MoviesDetailsCleaned es, en promedio, de un 2.4%, por otro lado, la diferencia en la carga de *MoviesCastingRaw es, en promedio, de un 5.8%.* Son bastante similares*.*
* ¿Cuál de las dos implementaciones (*Arraylist* y *Singlelinkedlist*) tiene mejor desempeño? Y ¿Por qué?: En base a los datos de la tabla la implementación Arraylist tiene un mejor desempeño; los tiempos de ejecución de los algoritmos de ordenamiento son casi 100 veces más rápidos. Esto podría deberse a que la función GetElement (y otras más) tiene un orden de crecimiento constante en los arraylist, contrariamente a lo que sucede en las single linked list, donde GetElement tiene un orden de crecimiento lineal.
* ¿Qué diferencias existen entre cargar los archivos de películas (*MoviesDetailsCleaned*) y elenco (*MoviesCastingRaw*)?, ¿Por qué se presentan estas diferencias?: En promedio, el tiempo de carga de *MoviesCastingRaw* es un 53% del tiempo de carga de *MoviesDetailsCleaned.* Es decir*, MoviesCastingRaw* se carga casi dos veces más rápido. Esto podría deberse a que *MoviesDetailsCleaned* tiene más columnas que *MoviesCastingRaw.*
* ¿Qué diferencias en el desempeño se observan entre los tres algoritmos de ordenamiento?: Insertion sort y Selection sort no difieren mucho en tiempo, en promedio, hay una discrepancia de un 22%, por el contrario, Shell sort resulta, en promedio, 55 veces más rápido que los dos algoritmos de ordenamiento anteriormente mencionados.
* ¿Qué efectos tienen los dos tipos de lista en los tres algoritmos de ordenamiento?: Al cambiar a sinlge linked list los tiempos se multiplicaron, en promedio, por 109.