OBSERVACIONES DE LA PRACTICA

Andres Fernando Galvis Cod 201632930

Alejandro Palomino Vargas 2 Cod 201913699

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | AMD Ryzen 5 2600x | AMD Ryzen 5 3500U |
| Memoria RAM (GB) | 16GB RAM | 8GB RAM |
| Sistema Operativo | Windows 10 Pro 64 bit | Windows 10 Home 64 bit |

Tabla 1. Especificaciones de las máquinas para ejecutar las pruebas de rendimiento.

# **Maquina 1**

## **Resultados**

|  |  |  |
| --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 31.25 | 15.62 |
| 2000 | 62.50 | 62.50 |
| 4000 | 140.62 | 140.62 |
| 8000 | 265.62 | 265.62 |
| 16000 | 656.25 | 661.40 |
| 32000 | 1383.54 | 1391.00 |
| 64000 | 3354.00 | 3391.00 |
| 128000 | 9479.10 | 9516.00 |
| 256000 | se demoró mucho | se demoró mucho |
| 512000 | se demoró mucho | se demoró mucho |

Tabla 2. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.

|  |  |  |
| --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 2489.58 | 2484.37 |
| 2000 | 10812.50 | 10859.37 |
| 4000 | 48671.88 | 48781.25 |
| 8000 | 191671.88 | 192265.62 |
| 16000 | se demoró mucho | se demoró mucho |
| 32000 | se demoró mucho | se demoró mucho |
| 64000 | se demoró mucho | se demoró mucho |
| 128000 | se demoró mucho | se demoró mucho |
| 256000 | se demoró mucho | se demoró mucho |
| 512000 | se demoró mucho | se demoró mucho |

Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Merge sort | Es mas eficiente | Es menos eficiente |
| Quick sort | Es mas eficiente | Es menos eficiente |

Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

# **Maquina 2**

## **Resultados**

|  |  |  |
| --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 46.87 | 31.25 |
| 2000 | 78.12 | 78.12 |
| 4000 | 156.25 | 171.87 |
| 8000 | 328.12 | 328.12 |
| 16000 | 812.50 | 734.37 |
| 32000 | 1765.62 | 1703.12 |
| 64000 | 4000.00 | 4140.62 |
| 128000 | 11203.12 | 11062.50 |
| 256000 | se demoró mucho | se demoró mucho |
| 512000 | se demoró mucho | se demoró mucho |

Tabla 2. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.

|  |  |  |
| --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 2718.75 | 2781.25 |
| 2000 | 11453.12 | 12000.00 |
| 4000 | 51703.12 | 51687.50 |
| 8000 | 215171.87 | 211234.37 |
| 16000 | se demoró mucho | se demoró mucho |
| 32000 | se demoró mucho | se demoró mucho |
| 64000 | se demoró mucho | se demoró mucho |
| 128000 | se demoró mucho | se demoró mucho |
| 256000 | se demoró mucho | se demoró mucho |
| 512000 | se demoró mucho | se demoró mucho |

Tabla 3. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Merge sort | Es mas eficiente | Es menos eficiente |
| Quick sort | Es mas eficiente | Es menos eficiente |

Tabla 4. Comparación de eficiencia de acuerdo con los algoritmos de ordenamientos y estructuras de datos utilizadas.

## **Graficas**

# **Preguntas de análisis**

1. ¿El comportamiento de los algoritmos es acorde a lo enunciado teóricamente?

Teóricamente, el comportamiento promedio de ambos ordenamientos recursivos tiene una complejidad temporal de **nlogn.** Este comportamiento, entre lineal y cuadrático, forma una parábola no tan inclinada. El cual es justamente el crecimiento que vemos en todas las gráficas.

1. ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?

A pesar de la diferencia en el procesador, y de que una máquina tiene el doble de memoria RAM que la otra, no existen diferencias notables entre los tiempos en cada máquina.

1. De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?

No existen diferencias apreciables entre los resultados en cada máquina.

1. ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?

Si sólo tenemos en cuenta los tiempos, claramente un Array List supera enormemente en eficiencia a una lista encadenada. Ya que, esta primera estructura, fue capaz de ordenar un gran conjunto de 128000 datos en únicamente 9 u 11 segundos. Mientras que en la lista encadenada sólo se pudieron ordenar 8000.

1. Para el caso analizado de ordenamiento de los videos, teniendo en cuenta los resultados de tiempo reportados por todos los algoritmos de ordenamiento estudiados (iterativos y recursivos), proponga un ranking de los algoritmos de ordenamiento (de mayor eficiencia a menor eficiencia en tiempo) para ordenar la mayor cantidad de videos.

De mayor a menor eficiencia, escogiendo un Array List como estructura de datos, obtenemos

* Merge Sort
* Quick Sort
* Shell Sort
* Insertion Sort
* Selection Sort