OBSERVACIONES DE LA PRACTICA

Juan Sebastian Grillo Cod 202116631

Estudiante 2 Cod XXXX

# **Ambientes de pruebas**

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | AMD Ryzen 5 5600x 6-Core Processor  3.70 GHz |  |
| Memoria RAM (GB) | 8 GB |  |
| Sistema Operativo | Windows 10 Pro |  |

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

# **Maquina 1**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 768 | 1656.25 | 78.25 | 78.25 | 78.25 |
| 100.00% | 10000 | 285906.25 | 2046.75 | 8859.37 | 1343.75 |

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% | 768 | 12875.0 | 656.25 | 625.0 | 140.62 |
| 100.00% | 10000 | Se espero 20 minutos y no cargo | 190453.125 | 334668.75 | 15875.0 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 285906.25 | No cargo |
| *Shell Sort* | 2046.75 | 190453.125 |
| *Merge Sort* | 1343.75 | 15875.0 |
| *Quick Sort* | 8859.37 | 334668.75 |

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

# **Maquina 2**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% |  |  |  |  |  |
| 100.00% |  |  |  |  |  |

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Porcentaje de la muestra [pct]** | **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 0.50% |  |  |  |  |  |
| 100.00% |  |  |  |  |  |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* |  |  |
| *Shell Sort* |  |  |
| *Merge Sort* |  |  |
| *Quick Sort* |  |  |

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

# **Preguntas de análisis**

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

RTA:

Según lo teórico si se es acorde, ya que como se aprecia, el algoritmo MergeSort fue el mas rápido en ordenar, luego el Shell short , quiksort y por ultimo insertion Sort, esto estaba especificado en las guía del laboratorio 4 y los apuntes de clase

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

RTA:

No se reporto compañero

1. De existir diferencias, ¿a qué creen que se deben?

RTA:

No se reportó compañero

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

RTA:

La estructura de datos según lo analizado, por bastante margen de diferencia, vendría siendo el arreglo ARRAY\_LIST, el cual también ya se predecía ese resultado.

1. Teniendo en cuenta las pruebas de tiempo de ejecución por todos los algoritmos de ordenamiento estudiados (iterativos y recursivos), proponga un ranking de los mismo de mayor eficiencia a menor eficiencia en tiempo para ordenar la mayor cantidad de obras de arte.

RTA:

1puesto: Merge Sort

2puesto: Shell Sort

3puesto: Quick Sort

4puesto: Insertion Sort