OBSERVACIONES DE LA PRACTICA

Julio Mario Rodriguez 202020236

Estudiante 2 Cod XXXX

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz | ADM Ryzen 5 3500u  Whit Radeon vega mobile Gfx 2.10 GHz |
| Memoria RAM (GB) | 16 GB | 9,95 GB |
| Sistema Operativo | Windows 10 | Windows 10 home 64 bits |

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

# **Maquina 1**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 107312,50 | 102609,38 | 5328,13 | 4421,88 | 515,63 |
| 2000 | 883843,75 | 778218,75 | 27593,75 | 19593,75 | 2234,38 |
| 4000 | 7025343,75 | 6504625,00 | 128515,63 | 84453,13 | 7640,63 |
| 8000 | - | - | 549156,25 | 348625,20 | 30937,50 |
| 16000 | - | - | 2670343,75 | 1601750,00 | 124500,00 |
| 32000 | - | - | - | - | 562734,38 |
| 64000 | - | - | - | - | 2051671,88 |
| 128000 | - | - | - | - | - |
| 256000 | - | - | - | - | - |
| 512000 | - | - | - | - | - |

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 103484,38 | 91656,25 | 5156,25 | 3953,13 | 500,00 |
| 2000 | 840375,00 | 737765,63 | 23750,00 | 17828,13 | 2000,00 |
| 4000 | 6879015,63 | 6486203,13 | 131687,50 | 95421,88 | 8578,13 |
| 8000 | - | - | 617671,88 | 322937,50 | 35156,25 |
| 16000 | - | - | 2707265,63 | 1610640,63 | 126515,63 |
| 32000 | - | - | - | - | 571421,88 |
| 64000 | - | - | - | - | 2311187,50 |
| 128000 | - | - | - | - | - |
| 256000 | - | - | - | - | - |
| 512000 | - | - | - | - | - |

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 | X |  |
| Quick sort |  | X |

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

## **Graficas**

* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 1.**
  + Comparación de rendimiento ARRAYLIST.
  + Comparación de rendimiento LINKED\_LIST.
  + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.
  + Comparación de rendimiento para Shell Sort.
  + Comparación de rendimiento para MergeSort.
  + Comparación de rendimiento para QuickSort.

# **Maquina 2**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 48718,75 | 43718,75 | 2546,46 | 2093,75 | 250,00 |
| 2000 | 373453,12 | 406031,25 | 10859,37 | 9453,12 | 1015,62 |
| 4000 | 2943031,25 | 3249468,75 | 51875,00 | 38187,50 | 3718,75 |
| 8000 |  |  | 266750,00 | 168828,12 | 14937,50 |
| 16000 |  |  | 1261859,00 | 741781,25 | 61562,00 |
| 32000 |  |  |  |  |  |
| 64000 |  |  |  |  |  |
| 128000 |  |  |  |  |  |
| 256000 |  |  |  |  |  |
| 512000 |  |  |  |  |  |

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (LINKED\_LIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 50875,00 | 45390,62 | 2687,50 | 2062,50 | 234,37 |
| 2000 | 420984,00 | 376031,25 | 11031,25 | 9250,00 | 1015,62 |
| 4000 |  |  | 54078,12 | 37109,00 | 3687,50 |
| 8000 |  |  |  | 164671,87 | 15122,00 |
| 16000 |  |  |  | 721203,12 | 62031,25 |
| 32000 |  |  |  |  |  |
| 64000 |  |  |  |  |  |
| 128000 |  |  |  |  |  |
| 256000 |  |  |  |  |  |
| 512000 |  |  |  |  |  |

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 |  |  |
| Quick sort |  |  |

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

## **Graficas**

* Cinco gráficas generadas por los resultados de las pruebas de rendimiento en la **Maquina 2.**
  + Comparación de rendimiento ARRAYLIST.
  + Comparación de rendimiento LINKED\_LIST.
  + Comparación de rendimiento para Insertion Sort.
  + Comparación de rendimiento para Selection Sort.
  + Comparación de rendimiento para Shell Sort.
  + Comparación de rendimiento para MergeSort.
  + Comparación de rendimiento para QuickSort.

# **Preguntas de análisis**

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

R) En verdad las graficas difieren un poco de lo teórico pero puede ser a causa de la poca cantidad de datos que pueda moldear mas acertadamente la curva.

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

R) si, un cambio importante en tiempo empleado para realizar el ordenamiento.

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

R) En la cantidad de datos que pueden procesar, y curiosamente la segunda máquina que tiene menor capacidad procesa a más velocidad.

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

R) La Estructura de Datos que es la mejor considerando solo el tiempo sería el LINKED\_LIST.

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.

R)

1 – Quick Sort

2 – Merge Sort

3 – Shell Short

4 – Selection Sort

5 - Insertion Sort