OBSERVACIONES DE LA PRACTICA

Juan Esteban López cod 202020285

Estudiante 2 Cod XXXX

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel(R) core(TM) i5-6500 cpu 3.20GHz | AMD Ryzen 7 3700U  2.30GHz |
| Memoria RAM (GB) | 16 GB | 8 GB |
| Sistema Operativo | Windows 10 pro 64-bits | Windows 10 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 | 953.125 | 937.5 | 78.125 | 31.25 | 31.25 |
| 2000 | 4000 | 3859.375 | 140.625 | 78.125 | 62.5 |
| 4000 | 16734.375 | 15921.875 | 296.875 | 171.875 | 156.25 |
| 8000 | 69187.5 | 69221.875 | 697.5 | 312.5 | 328.125 |
| 16000 | 286718.75 | 292796.875 | 1578.125 | 671.875 | 703.125 |
| 32000 | 1159593.75 | 1185671.875 | 3859.375 | 1593.75 | 1500.0 |
| 64000 |  |  | 5750 | 3375.0 | 3656.25 |
| 128000 |  |  | 14359.375 | 7187.5 | 7171.875 |
| 256000 |  |  | 34218.75 | 16406.25 | 15234.375 |
| 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 | 365984.375 | 337953.125 | 11000 | 2265.625 | 296.875 |
| 2000 |  |  | 51953.125 | 12031.25 | 1156.25 |
| 4000 |  |  | 233265.625 | 51562.5 | 4875.00 |
| 8000 |  |  |  | 217953.125 | 19281.25 |
| 16000 |  |  |  |  | 77359.38 |
| 32000 |  |  |  |  | 319453.13 |
| 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 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 | 640.625 | 671.875 | 31.25 | 46.875 | 46.875 |
| 2000 | 2640.625 | 2625 | 78.125 | 125 | 109.375 |
| 4000 | 10531.25 | 11906.25 | 187.5 | 296.875 | 250 |
| 8000 | 45062.5 | 43140.625 | 437.5 | 546.875 | 500 |
| 16000 | 183062.5 | 195375 | 1093.75 | 1218.75 | 1406.25 |
| 32000 | 709890.625 | 769625 | 2609.375 | 2750 | 2968.75 |
| 64000 |  |  | 5593.75 | 5953.125 | 6718.75 |
| 128000 |  |  | 14109.375 | 13281.25 | 11734.375 |
| 256000 |  |  | 33000.0 | 32365.625 | 29234.375 |
| 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 | 55781.25 | 51640.625 | 2765.625 | 3500 | 468.75 |
| 2000 | 464328.125 | 417343.75 | 14046.875 | 16812 | 1984.375 |
| 4000 |  |  | 65765.625 | 75750 | 5578.125 |
| 8000 |  |  | 294265.625 | 204156.25 | 22375 |
| 16000 |  |  |  |  | 88484.375 |
| 32000 |  |  |  |  | 370234.375 |
| 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?

Si es acorde a lo enunciado teóricamente.

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

Si hay diferencias en cuanto a la velocidad en la que cada máquina logra sus objetivos más no en la eficacia de cada algoritmo de ordenamiento y la estructura de datos utilizada.

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

La eficiencia más clara entre las dos máquinas es la memoria RAM de cada una, puesto que en la máquina que se logró la mayor rapidez tenía 16GB RAM y la otra máquina que fue más lenta solo tenía 8 GB RAM. Otra diferencia es la de procesadores, esta no la entiendo muy bien debido a que el procesador de la maquina más rápida es uno más viejo y de menor gama, pero con mayor rapidez mientras que el otro es de menor gama, pero de menor rapidez.

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

Si solo se tiene en cuenta la rapidez de ejecución la mejor estructura de datos es el ARRAY\_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.
2. Mergesort
3. Quicksort
4. Shellsort
5. Selectionsort
6. Insertionsort