OBSERVACIONES DE LA PRACTICA

Estudiante 1 Cod 202021673

Estudiante 2 Cod XXXX

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel® Core™ i7-9750H CPU @ 2.60Hz 2.59Hz | Intel® Core™ i7-10510U CPU @ 1.80 Hz 2.30Hz |
| Memoria RAM (GB) | 32.0 GB | 16.0 GB |
| Sistema Operativo | Windows 10 Pro-64 bits | Windows 10 Pro-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 | 652,30 | 1004,56 | 51,85 | 31,25 | 31,25 |
| 2000 | 2863,47 | 3867,25 | 86,20 | 62,5 | 62,5 |
| 4000 | 11827,36 | 12365,20 | 222,42 | 144,73 | 132,63 |
| 8000 | 46852,40 | 57289,60 | 589,65 | 317,86 | 274,79 |
| 16000 | 194568,79 | 217896,52 | 1204,80 | 604,17 | 593,75 |
| 32000 | 676895,85 | 718541,94 | 2506,54 | 1166,67 | 1218,75 |
| 64000 | Tiempo excedido | Tiempo excedido | 6008,52 | 2294,79 | 2645,84 |
| 128000 | Tiempo excedido | Tiempo excedido | 18235,14 | 6593,75 | 6713,54 |
| 256000 | Tiempo excedido | Tiempo excedido | 66312,69 | 13093,75 | 12119,79 |
| 512000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Out of range | Out of range |

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 | 94565,14 | 81965,32 | 4405,26 | 1609,38 | 390,625 |
| 2000 | 285697,32 | 256982,56 | 25718,75 | 8343,75 | 3281,25 |
| 4000 | 666128,10 | 606140,06 | 92437,50 | 42015,63 | 6187,5 |
| 8000 | Tiempo excedido | Tiempo excedido | 274695,24 | 340281,25 | 24828,13 |
| 16000 | Tiempo excedido | Tiempo excedido | 708965,33 | Tiempo excedido | 109281,25 |
| 32000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | 427500,00 |
| 64000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 128000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 256000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 512000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Out of range | Out of range |

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**

# **Maquina 2**

## **Resultados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tamaño de la muestra (ARRAYLIST)** | **Insertion Sort [ms]** | **Selection Sort [ms]** | **Shell Sort [ms]** | **Quick Sort [ms]** | **Merge Sort [ms]** |
| 1000 | 890,625 | 843,75 | 46,875 | 62,5 | 46,875 |
| 2000 | 2718,75 | 3390,625 | 125,0 | 93,75 | 93,75 |
| 4000 | 10671,875 | 12437,5 | 250,0 | 203,125 | 140,625 |
| 8000 | 43140,625 | 51671,875 | 546,87 | 328,125 | 390,625 |
| 16000 | 338484,375 | 228656,25 | 843,75 | 843,75 | 875,0 |
| 32000 | Tiempo excedido | Tiempo excedido | 1875,0 | 1484,375 | 1453,125 |
| 64000 | Tiempo excedido | Tiempo excedido | 4390,625 | 3234,375 | 2859,375 |
| 128000 | Tiempo excedido | Tiempo excedido | 10750,0 | 6578,125 | 6265,625 |
| 256000 | Tiempo excedido | Tiempo excedido | 26625,0 | 14437,5 | 13109,375 |
| 512000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |

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 | 55453,125 | 40187,193 | 2754,31 | 1906,25 | 390,625 |
| 2000 | 384254,193 | 364254,52 | 14325,2 | 8125,0 | 1390,625 |
| 4000 | Tiempo excedido | Tiempo excedido | 78716,7 | 430406,25 | 4500,0 |
| 8000 | Tiempo excedido | Tiempo excedido | 400540,3 | 176421,375 | 18828,125 |
| 16000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | 113406,25 |
| 32000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 64000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 128000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 256000 | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido | Tiempo excedido |
| 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**

# **Preguntas de análisis**

* ¿El comportamiento de los algoritmos es acorde a lo enunciado teóricamente?
  + Si, ya que como vimos el Insertion y el Selection es de ordenamiento de O(n^2) mientras que el Shell es O(n log ^ 2 n) , el Quick y Merge son de ordenamiento O(n log n), y en este caso lo ideal es O(n log n), y como vemos el Shell, Quick y Merge, son los que más parecido tiene con el ideal, y como también podemos ver en las gráficas el Shell, el Quick y Merge son los más indicados en esta situación.
* ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?
  + Al ejecutar el programa como tal no hay diferencias, pero al cargar archivos y tomar la muestra de datos, ahí si se logra ver una notoria diferencia en cuanto a la velocidad que se ordenan los datos.
* De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?
  + Por lo que se nota, las máquinas tienen diferentes procesadores, pero la principal y notoria diferencia es en la RAM de las máquinas, ya que la segunda máquina tiene 16GB de RAM, pero la primera máquina tiene 32GB de RAM, por lo que gracias a esto a la segunda máquina tiene una mayor velocidad a la hora de examinar esa muestra de datos.
* ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?
  + Como nos damos cuenta la Estrucura de Datos que cuenta con mayor velocidad de ejecución en el algoritmo es el arreglo o Array List y siendo más específicos el Shell Sort.
* 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.
  + Tomando en cuenta los datos que se ingresaron en las tablas y gráficas podemos concluir, que el ordenamiento de los algoritmos, (de mayor eficiencia a menor eficiencia en tiempo) queda en el siguiente orden:

1. Quick Sort– Array List
2. Merge Sort – Array List
3. Shell Sort – Array List
4. Selection Sort – Array List
5. Insertion Sort – Array List
6. Quick Sort – Linked List
7. Merge Sort – Linked List
8. Shell Sort – Linked List
9. Selection Sort – Linked List
10. Insertion Sort – Linked List