OBSERVACIONES DE LA PRACTICA

Estudiante 1 Cod XXXX

Martín Santiago Galván Castro 201911013

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 (Martín) |
| Procesadores |  | AMD Ryzen 3 3200G with Radeon Vega Graphics 3.60 GHz |
| Memoria RAM (GB) |  | 16 |
| Sistema Operativo |  | Windows 10 Pro-64-Bits |

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

# **Maquina 1**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 |  |  |  |
| 2000 |  |  |  |
| 4000 |  |  |  |
| 8000 |  |  |  |
| 16000 |  |  |  |
| 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) |
| 1000 |  |  |  |
| 2000 |  |  |  |
| 4000 |  |  |  |
| 8000 |  |  |  |
| 16000 |  |  |  |
| 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) |
| Insertion sort |  |  |
| Selection sort |  |  |
| Shell 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.

# **Maquina 2**

## **Resultados**

|  |  |  |  |
| --- | --- | --- | --- |
| Tamaño de la muestra (ARRAY\_LIST) | Insertion Sort (ms) | Selection Sort (ms) | Shell Sort (ms) |
| 1000 | 645.83 | 609.38 | 31.25 |
| 2000 | 2567.71 | 2375 | 78.12 |
| 4000 | 10427.08 | 9562.5 | 171.88 |
| 8000 | 42666.67 | 40890.62 | 406.25 |
| 16000 | 184203.12 | 159109.38 | 968.75 |
| 32000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 2156.25 |
| 64000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 5031.25 |
| 128000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 12171.88 |
| 256000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 28687.5 |
| 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) |
| 1000 | 30046.88 | 43984.38 | 1734.38 |
| 2000 | 241250 | 344046.88 | 7406.25 |
| 4000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 36218.75 |
| 8000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | 165984.38 |
| 16000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) |
| 32000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) |
| 64000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) |
| 128000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) |
| 256000 | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) | Exceso de tiempo (> 10 min) |
| 512000 | - | - | - |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| Insertion sort | Mas eficiente | Menos eficiente |
| Selection sort | Mas eficiente | Menos eficiente |
| Shell sort | Mas eficiente | Menos eficiente |

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.

# **Preguntas de análisis**

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

Por el lado de la maquina 2. El análisis de los datos para los algoritmos de insertion sort y selection sort se ven en parte limitados por la disponibilidad de datos. Adicionalmente, se ve limitado el análisis por el catálogo de líneas de tendencia de Excel. Sin embargo, se pueden establecer similitudes y diferencias con los datos teóricos.

Por parte de los algoritmos de selección selection sort e incertion sort, se observa que sus crecimientos en tiempos de ejecución se ven limitados por funciones cuadráticas. Por lo que se establece que, para este caso, se comportan acorde al peor caso de cada función. Por el lado del algoritmo de Shell sort, se evidencia que es el que mejor rendimiento tiene para ambos tipos de estructuras de datos. Adicionalmente, el algoritmo Shell sort con la estructura de datos array list, es el único que logra organizar todos los elementos en tiempos aceptables.

Por parte en como se comportan, se estima que se comportan como O(nlog(n). Ya que, al compararlas con funciones lineales, son parecidas, y para datos mayores, el comportamiento del tiempo de ejecución tiende a ser mayor a una función lineal.

1. ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?
2. De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?
3. ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?