OBSERVACIONES DE LA PRACTICA

# Daniel Rodriguez - ds.rodriguezf1@uniandes.edu.co - 202014760

# Santiago Forero - s.forerog2@uniandes.edu.co – 202111446

# **Ambientes de pruebas**

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx 2.30 GHz | Intel(R) Core™ ie-10210U CPU 1.60GHz 2.11 GHz1.60 GHz |
| Memoria RAM (GB) | 16,0 GB | 8,00 GB |
| Sistema Operativo | Windows 10 - 64 bits | Windows 10 - 64 bits |

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% | 50 | 0 | 0 | 0 | 0 |
| 100.00% | 10000 | 97234,37 | 996,09 | 2286,46 | 648,44 |

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% | 50 | 20,73 | 10,42 | 5,21 | 0 |
| 100.00% | 10000 | None | None | None | 30479,17 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 97234,37 | None |
| *Shell Sort* | 996,09 | None |
| *Merge Sort* | 2286,46 | None |
| *Quick Sort* | 648,44 | 30479,17 |

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% | 50 | 0 | 0 | 0 | 0 |
| 100.00% | 10000 | 102682,46 | 1012,45 | 2498,87 | 675,14 |

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% | 50 | 23,26 | 12,26 | 7,15 | 0 |
| 100.00% | 10000 | None | None | None | None |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 102682,46 | None |
| *Shell Sort* | 1012,45 | None |
| *Merge Sort* | 2498,87 | None |
| *Quick Sort* | 675,14 | None |

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?

Sí, se puede ver en el Excel como las gráficas esperadas coinciden con los resultados de funciones, al igual que las funciones estimadas como O(nlogn) o O(n^2)

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

Sí, en la segunda máquina los tiempos son ligeramente mayores

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

Se deben principalmente a dos cosas, la primera es que cada computador tiene potencia y memoria RAM distinta, por lo que las iteraciones se realizaron más rápido en un que en el otro. Y la segunda posible razón es la cantidad de programas abiertos o cuanto tiempo lleva encendido el computador, ya que esto afecta directamente a la RAM disponible, ya que aunque se intentó cerrar todas las pestañas abiertas, algunos procesos seguían en ejecución; y es por esto que en varias ocasiones no se pudo terminar el proceso de recursión.

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

La estructura de datos que mejor funciona es el arreglo o “ARRAY\_LIST” como se pudo observar el los resultados. Esto debido a que los algoritmos de ordenamientos funcionan con base en los intercambios, algo que toma menos tiempo en los arreglos que en las listas encadenadas.

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.

**Mayor eficiencia**

* Merge Sort
* Shell Sort
* Quick Sort
* Insertion Sort

**Menor eficiencia**

**NOTA: Al momento de intentarlo con el 100% de la Lista encadenada muchas veces no se terminó el proceso, a eso se refiere por None. De todas formas, en el Excel se encuentran anotados los resultados del procesamiento con todos los porcentajes que eran requeridos.**