OBSERVACIONES DE LA PRACTICA

Sofía Escobar 202021047

Carolina Chinchilla 202011842

# **Ambientes de pruebas**

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | AMD Ryzen 5 3600X 6-core | AMD Ryzen 3 3200U with Radeon Vega Mobile Gfx 2.60 GHz |
| Memoria RAM (GB) | 8 GB | 8,00 GB |
| Sistema Operativo | Windows 1064 bits | Sistema operativo de 64 bits, procesador x64 |

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]** |
| Small | 295.00 | 31.25 | 46.88 | 112.28 | 46.88 |
| 10.00% | 15008.00 | 1593.75 | 1625.00 | 953.13 | 4000.00 |

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]** |
| Small | 295.00 | 125.00 | 140.63 | 109.38 | 62.50 |
| 10.00% | 5960.00 | 105328.13 | 80578.13 | 139265.63 | 13234.38 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 2 | 2 |
| *Shell Sort* | 3 | 1 |
| *Merge Sort* | 4 | 3 |
| *Quick Sort* | 1 | 4 |

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]** |
| Small% | 295 | 125 | 78.25 | 78.125 | 62.5 |
| 10.00% | 5960 | 3031.25 | 3046.875 | 8546.875 | 1765.625 |

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]** |
| Small% | 295 | 265.625 | 250.0 | 203.125 | 93.75 |
| 10.00% | 5960 | 193015.625 | 145671.875 | 293203.125 | 12500.0 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | Mas eficiente | Menos eficiente |
| *Shell Sort* | Más eficiente | Menis eficiente |
| *Merge Sort* | Más eficiente | Menos eficiente |
| *Quick Sort* | Más eficiente | Menos eficente |

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 dado que, teniendo en cuenta lo visto en clase, dos de las formas más efectivas de organizar datos son Quick sort y Merge sort, y como se puede observar en los registros se cumplió que fueron los métodos que menor tiempo tomaron, siendo el Merge sort el que obtuvo mejores tiempos. Cabe destacar que, aunque Insertion sort funciona mejor con datos de pequeña escala aun así obtuvo peores tiempos en cuanto de tomó como muestra small.

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

Sí, los tiempos de ejecución registrados tienden a variar, registrándose mayores tiempos en la máquina 2.

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

La diferencia entre la máquina 1 y la máquina 2 son los procesadores, dado que la máquina 1 cuenta con 6-core y la máquina 2 2-core, los cuales son los núcleos de un procesador, está diferencia radica en que entre mayor cantidad de núcleos mayor cantidad ciclos de instrucciones independientes se pueden realizar, por lo cual el rendimiento de la máquina 1 es mayor.

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

Si se tiene solamente en cuenta los tiempos de ejecución obtenidos en el laboratorio entonces la mejor estructura de datos sería ARRAY-LIST, la cual tiende a ser casi 100% más efectiva al momento medir el tiempo de ejecución de funciones.

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.
   * 1. Merge sort, fue el que tuvo mejores tiempos de ordenamiento.
     2. Quick sort, en la máquina uno se observó que al procesar los datos como ARRAY-LIST es más demorado que Shell sort, sin embargo, en los demás campos obtuvo mejores resultados.
     3. Shell sort
     4. Insertio sort, este método de ordenamiento fue el que más tiempo se demoro en organizar los ítems.