OBSERVACIONES DE LA PRACTICA

Máquina 1 - Tomás la Rotta - 202021354

Máquina 2 - Daniela Espinosa - 202022615

# **Ambientes de pruebas**

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99 GHz | AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx 2.10 GHz |
| Memoria RAM (GB) | 8,00 GB (7,71 GB utilizable) | 8,00 GB (5,95 GB usable) |
| 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 | 31.25 | 31.25 | 15.625 | 15.625 |
| 10.00% | 1000 | 33437.5 | 9906.25 | 5781.25 | 281.25 |

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 | 0.0 | 0.0 | 0.0 | 0.0 |
| 10.00% | 1000 | 34687.5 | 8671.87 | 5875.0 | 218.75 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 16734.375 | 17343.75 |
| *Shell Sort* | 4968.75 | 4335.93 |
| *Merge Sort* | 148.43 | 109.37 |
| *Quick Sort* | 2898.43 | 2937.5 |

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 | 15.625 | 15.625 | 0.0 | 0.0 |
| 10.00% | 1000 | 11518.0 | 1750.0 | 1556.25 | 78.125 |

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 | 0.0 | 0.0 | 0.0 | 0.0 |
| 10.00% | 1000 | 11765.625 | 1875.0 | 1375.0 | 85.75 |

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

|  |  |  |
| --- | --- | --- |
| Algoritmo | Arreglo (ARRAYLIST) | Lista enlazada (LINKED\_LIST) |
| *Insertion Sort* | 5766.81 | 5882.81 |
| *Shell Sort* | 882.81 | 937.5 |
| *Merge Sort* | 778.125 | 687.5 |
| *Quick Sort* | 39.06 | 42.87 |

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?

Después de realizar distintas pruebas al código del reto, podemos evidenciar mejor el tiempo que se demora cada algoritmo de ordenamiento y, efectivamente, confirmar lo visto teoricamente. Esto se puede ver por la diferencia de tiempos de cada uno de los algoritmos, siendo el Merge Sort el más veloz y por lo tanto el más eficiente. Seguido por Quick Sort, Shell Sort y, por último, Insertion Sort.

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

Si, hay una diferencia considerable ya que la maquina 2 es más veloz que la maquina 1.

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

Seguramente las diferencias se deben al tipo de procesador de las maquinas y a la memoria RAM utilizable, pues la maquina 2 tiene mejor procesador y más memoria RAM utilizable.

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 funciona mejr según los tiemposde ejecución es la SINGLE LINKED ya que sus tiempos son menores comparados con los de ARRAY LIST.

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.

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