OBSERVACIONES DE LA PRACTICA

Estudiante 1 Cod XXXX

Estudiante 2 Cod XXXX

# Máquina 1 Máquina 2

|  |  |  |
| --- | --- | --- |
| **Procesadores** | Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz | Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz |
| **Memoria RAM (GB)** | 16GB | 8,00 GB |
| **Sistema Operativo** | Windows 10 64-bit | Windows 10 64-bit |

*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

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]** | **Quick**  **Sort**  **[ms]** | **Merge**  **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)**

|  |  |  |
| --- | --- | --- |
| **Merge sort** |  |  |
| **Quick 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.**

o Comparación de rendimiento ARRAYLIST. o Comparación de rendimiento LINKED\_LIST. o Comparación de rendimiento para Insertion Sort. o Comparación de rendimiento para Selection Sort. o Comparación de rendimiento para Shell Sort. o Comparación de rendimiento para MergeSort. o Comparación de rendimiento para QuickSort.

# Maquina 2

## Resultados

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tamaño de la muestra Insertion (ARRAYLIST) Sort [ms]** | **Selection Sort [ms]** | **Shell**  **Sort**  **[ms]** | **Quick**  **Sort**  **[ms]** | **Merge**  **Sort**  **[ms]** |

1000

2000

4000

8000

16000

32000

64000

128000

256000

512000

*Tabla 5. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.*

### Tamaño de la muestra Sort [ms]Insertion Sort [ms]Selection Shell[Sortms] Quick[Sortms] Merge[Sortms] (LINKED\_LIST)

1000

2000

4000

8000

16000

32000

64000

128000

256000

512000

*Tabla 6. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación lista enlazada.*

### Algoritmo Arreglo (ARRAYLIST) Lista enlazada

**(LINKED\_LIST)**

|  |  |  |
| --- | --- | --- |
| **Merge sort** |  |  |
| **Quick sort** |  |  |

*Tabla 7. 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. o Comparación de rendimiento LINKED\_LIST.
* Comparación de rendimiento para Insertion Sort. o Comparación de rendimiento para Selection Sort. o Comparación de rendimiento para Shell Sort. o Comparación de rendimiento para MergeSort. o Comparación de rendimiento para QuickSort.

# Preguntas de análisis

1. ¿El comportamiento de los algoritmos es acorde a lo enunciado teóricamente?
2. ¿Existe alguna diferencia entre los resultados obtenidos al ejecutar las pruebas en diferentes máquinas?
3. De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?
4. ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?
5. 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.