

OBSERVACIONES DE LA PRACTICA

Santiago Tenjo 202113965
Jaider Rincón 202115689

	Máquina 1	Máquina 2
Procesadores	Intel® Core™ i7-9700 CPU @3.00GHz 3.00GHz	AMD Ryzen 5 4500U 2.38 GHz
Memoria RAM (GB)	8.00GB	7.56 GB
Sistema Operativo	Windows 10 pro 64bitz	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 (ARRAY_LIST)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]	Quick Sort [ms]	Merge Sort [ms]
0.50%	44.96	596.28	618.57	27.58	19.76	23.08
5.00%	343.87	34192.30	34700.92	285.07	205.52	190.40
10.00%	649.31	94870.75	97033.63	536.79	407.17	322.58
20.00%	1197.13	247727.79	243166.56	979.98	722.78	547.34
30.00%	1708.01	410937.80	401026.01	1339.55	972.16	726.52
50.00%	2603.55			1715.06	1404.97	929.77
80.00%	3725.35			2127.51	1912.82	1194.96
100.00%	4373.84			2623.75	1964.07	1271.74

Tabla 2. Comparación de tiempos de ejecución para los ordenamientos iterativos en la representación arreglo.

Porcentaje de la muestra [pct]	Tamaño de la muestra (LINKED_LIST)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]	Quick Sort [ms]	Merge Sort [ms]
0.50%	44.47	60987.99	55615.78	2345.03	1797.81	247.05
5.00%	353.16			232940.72	152708.52	13189.23
10.00%	681.07				521813.13	37638.07
20.00%	1231.42					
30.00%	1777.58					156862.73
50.00%	2836.01					
80.00%	3905.65					
100.00%	4611.79					

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**.
 - 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.
 - Comparación de rendimiento para MergeSort.
 - Comparación de rendimiento para QuickSort.

Maquina 2

Resultados

Porcentaje de la muestra [pct]	Tamaño de la muestra (ARRAY_LIST)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]	Quick Sort [ms]	Merge Sort [ms]
0.50%	101.52	79.25	93.99	76.89	68.61	73.17
5.00%	732.95	699.44	708.46	499.55	478.73	497.46
10.00%	1353.66	1505.01	1394.08	966.42	957.02	1033.86
20.00%	2789.66	2760.76	1947.66	1902.8	1873.05	2015.29
30.00%	4038.47	3974.47	2904.19	2709.6	2814.00	2739.94
50.00%	6771.12	6851.30	4868.53	4805.2	4683.87	4792.87
80.00%	10497.67	11661.62	7319.91	7608.9	7435.18	7421.33
100.00%	13415.78	17555.08	9781.25	9146.9	9593.68	8752.24

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

Porcentaje de la muestra [pct]	Tamaño de la muestra (LINKED_LIST)	Insertion Sort [ms]	Selection Sort [ms]	Shell Sort [ms]	Quick Sort [ms]	Merge Sort [ms]
0.50%	72.14					
5.00%	490.69					
10.00%	957.10					
20.00%	1863.99					
30.00%	2904.36					
50.00%	4933.97					
80.00%	7150.55					
100.00%	9224.39					

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.

- Comparación de rendimiento LINKED_LIST.
- Comparación de rendimiento para Insertion Sort.
- Comparación de rendimiento para Selection Sort.
- Comparación de rendimiento para Shell Sort.
- Comparación de rendimiento para MergeSort.
- Comparación de rendimiento para QuickSort.

Preguntas de análisis

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

Si teoricamente, los ordenamientos como shell, quick y merge son los que tienen a ejecutarse más rápido.

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

Si existen diferencias notables en los resultados que exceden los 10 minutos y los que no.

3) De existir diferencias, ¿A qué creen ustedes que se deben dichas diferencias?

En este caso las diferencias son más difíciles de notar ya que los datos tienden a ser más parecidos, pero sin lugar a dudas esto se debe al procesador y a la optimización de cada computador.

4) ¿Cuál Estructura de Datos es mejor utilizar si solo se tiene en cuenta los tiempos de ejecución de los algoritmos?

Array_List

5) Para el caso analizado de ordenamiento de los artistas, teniendo en cuenta los resultados de tiempo reportados por todos los algoritmos de ordenamiento (iterativos y recursivos), proponga un ranking de los algoritmos de ordenamiento (de mayor eficiencia a menor (en tiempos de ejecución) para ordenar la mayor cantidad de artistas.

Quick

Merge

Shell

Selection

Insertion