

# OBSERVACIONES DE LA PRACTICA

Raul Santiago Insuasty - 202015512

Juan Esteban Currea

## Ambientes de pruebas

	Máquina 1	Máquina 2
Procesadores	AMD RYZEN 5 2500U	Intel Core i5
Memoria RAM (GB)	16 GB	16 GB
Sistema Operativo	Windows 10 Home 64 bits	Windows 10 Home 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%	294	93.75	15.63	15.63	15.63
100.00%	13418	178,468.75	968.75	4218.75	734.38

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%	294	1000	109.38	140.63	15.63
100.00%	13418	N.T.	N.T.	N.T.	41531.25

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

Algoritmo	Arreglo (ARRAYLIST)	Lista enlazada (LINKED_LIST)
Insertion Sort	$O(n)$	$O(n^2)$
Shell Sort	$O(n)$	$O(n^{3/2})$
Merge Sort	$O(\log(n))$	$O(\log(n))$
Quick Sort	$O(n \log(n))$	$O(n^2)$

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%	294	95,73	18,2	17,9	14,4
100.00%	14218	182,89	940,4	4129,34	734.45

Tabla 5. 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%	294	987,8	107.24	141.52	15.34
100.00%	14218	N.T	N.T	N.T	40321.25

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

Algoritmo	Arreglo (ARRAYLIST)	Lista enlazada (LINKED_LIST)
<i>Insertion Sort</i>	$O(n)$	$O(n^2)$
<i>Shell Sort</i>	$O(n)$	$O(n^{3/2})$
<i>Merge Sort</i>	$O(\log(n))$	$O(\log(n))$
<i>Quick Sort</i>	$O(n \log(n))$	$O(n^2)$

Tabla 7. 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?

Si es posible distinguir una diferencia notable entre los algoritmos recursivos y los iterativos, sobre todo al momento de guardar datos en una lista enlazada, pues merge fue el único capaz de ejecutarse en nuestro computador con el archivo 10%.

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

No, las máquinas son muy similares y los cambios en el tiempo fueron insignificantes.

- 3) De existir diferencias, ¿a qué creen que se deben?

Las mínimas diferencias pueden deberse al procesador de cada máquina.

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

Indiscutiblemente Arraylist, pues por lo menos se pudo calcular el tiempo usando todos los algoritmos, tanto recursivos como iterativos.

- 5) 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.

El mejor es MergeSort superando por mucho a los demás, luego seguiría shell sort y quick sort que son parejos, y el peor de todos es Insertion Sort, que es demasiado ineficiente y lento.