

## Trabajo Práctico – Algoritmos de búsqueda y ordenamiento.

Presentado por:

Israel Garcia Moscoso – [isragadiel@gmail.com](mailto:isragadiel@gmail.com)

Juan Esteban Gelos – [juan\\_gelos@yahoo.com](mailto:juan_gelos@yahoo.com)

Materia: Programación I

Profesor: Ariel Enferrel

Tutor: Ramiro Hualpa

Fecha de Entrega: 9 de junio de 2025

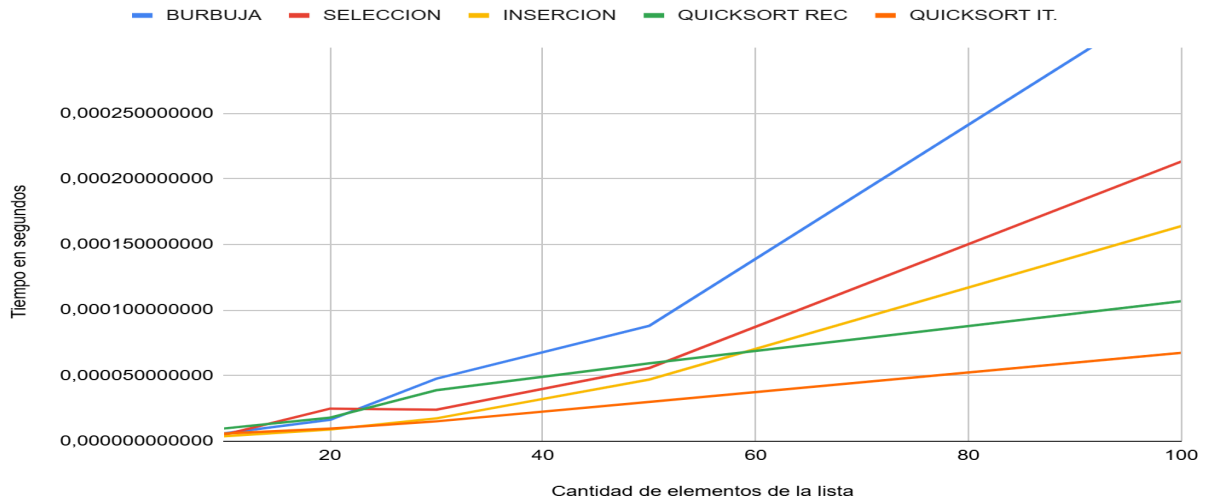
---

### Anexo 1: Análisis del Tiempo de respuesta Algoritmos de Ordenamiento

Probamos los algoritmos de ordenamiento por el método Burbuja, Selección, Inserción, Quicksort recursivo y Quicksort iterativo, en todos los casos sobre las mismas listas de elementos (números).

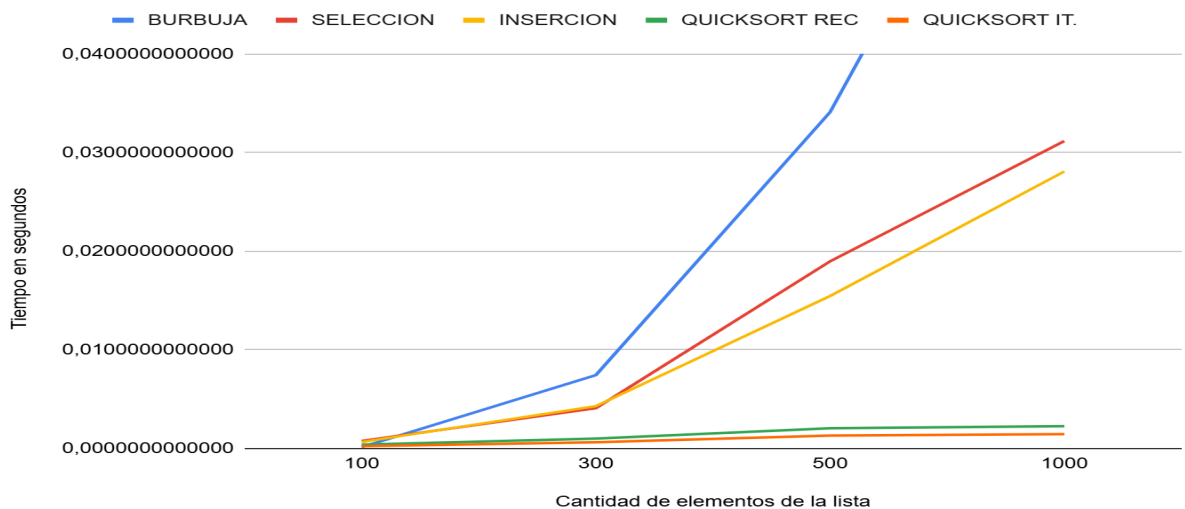
#### Listas de menos de 100 elementos

MÉTODO	TAMAÑO DE LA LISTA				
	10	20	30	50	100
BURBUJA	0,0000059000	0,0000162999	0,0000477000	0,0000880000	0,0003437999
SELECCION	0,0000048000	0,0000248000	0,0000239999	0,0000558000	0,0002134000
INSERCIÓN	0,0000036999	0,0000089000	0,0000173999	0,0000470000	0,0001641001
QUICKSORT REC	0,0000096001	0,0000179000	0,0000389001	0,0000594000	0,0001068000
QUICKSORT IT.	0,0000057999	0,0000096000	0,0000151000	0,0000299000	0,0000674000



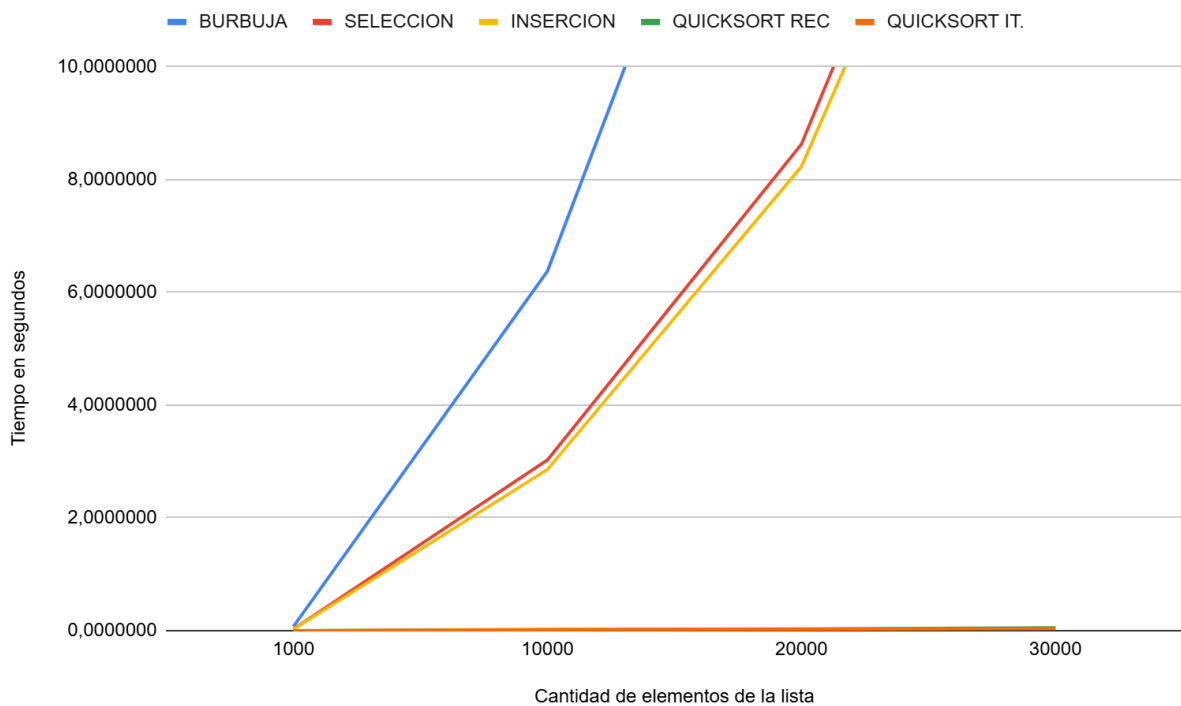
### Listas de 100 a 1000 elementos

	TAMAÑO DE LA LISTA			
MÉTODO	100	300	500	1.000
BURBUJA	0,0001222399994	0,0074273999780	0,0341003999347	0,0772568000248
SELECCION	0,0007437999593	0,0040787000908	0,0189671999542	0,0311579999980
INSERCIÓN	0,0006448203818	0,0042840000242	0,0154423000058	0,0280750999227
QUICKSORT REC	0,0003795999800	0,0009842000436	0,0020457999781	0,0022469999967
QUICKSORT IT.	0,0002216999419	0,0006249999860	0,0013006000081	0,0014348000987



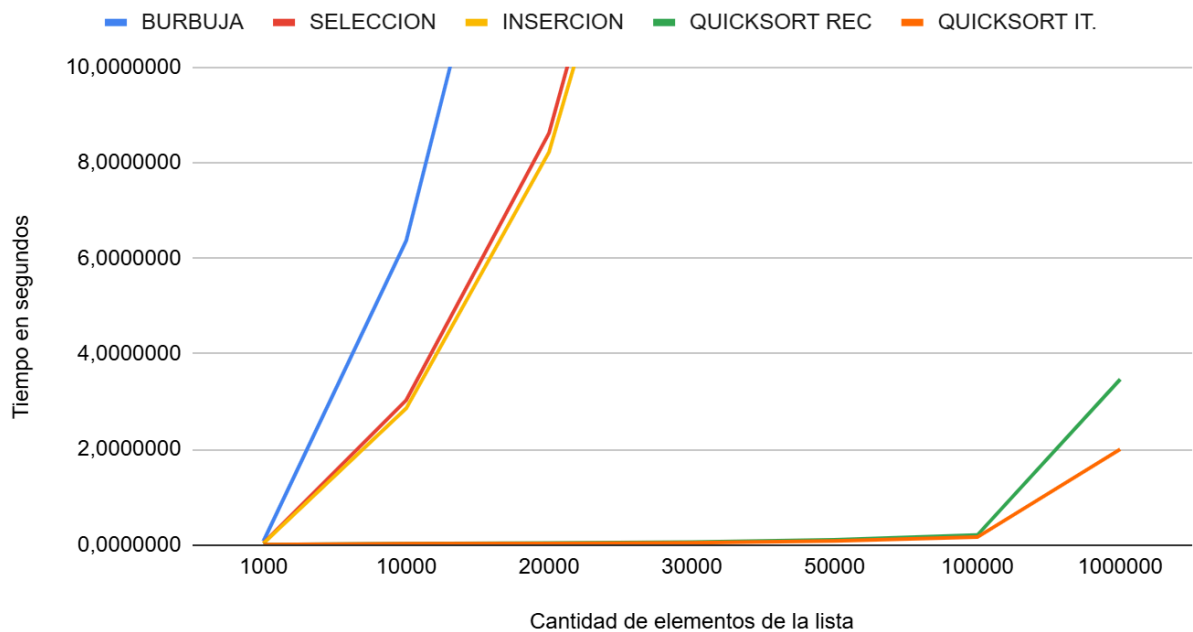
## Listas de 1.000 a 30.000 elementos

	TAMAÑO DE LA LISTA			
MÉTODO	1.000	10.000	20.000	30.000
BURBUJA	0,0772568000248	6,3726919000037	18,2252849000050	40,5317958999658
SELECCION	0,0311579999980	3,0281218999298	8,6251751999371	19,4858027000445
INSERCIÓN	0,0280750999227	2,8592969999881	8,2272669000085	18,5135035999119
QUICKSORT REC	0,0022469999967	0,0217719000066	0,0333637999138	0,0530365000013
QUICKSORT IT.	0,0014348000987	0,0180687999818	0,0253322999924	0,0384602999547



## Listas de 1.000 a 1.000.000 elementos

	TAMAÑO DE LA LISTA						
MÉTODO	1.000	10.000	20.000	30.000	50.000	100.000	1.000.000
BURBUJA	0,0772568	6,3726919	18,2252849	40,5317959			
SELECCION	0,0311580	3,0281219	8,6251752	19,4858027	56,4327347		
INSERCIÓN	0,0280751	2,8592970	8,2272669	18,5135036	54,8166085		
QUICKSORT REC	0,0022470	0,0217719	0,0333638	0,0530365	0,1034973	0,202242	3,465860
QUICKSORT IT.	0,0014348	0,0180688	0,0253323	0,0384603	0,0760014	0,157293	1,997083



**Algunos print de pantallas de la respuesta de los algoritmos.**

Para 100 elementos

```
ordenamiento.py U X
ordenamiento.py > ...
1 import random
2 import timeit
3 from generador_lista import generar_lista_desde_txt
4
5 # Generar una lista aleatoria de tamaño 1000
6 #archivo="personas1.txt"
7 #data=generar_lista_desde_txt(archivo)
8 #random.shuffle(data)
9 data=[random.randint(1, 1000000000) for _ in range(100)]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

[BURBUJA] Tiempo de ejecución para la lista pequeña fue de: 1.3399985618889332e-05 segundos
[BURBUJA] Tiempo de ejecución para lista grande: 0.0003437999403104186 segundos

[SELECCION] Tiempo para una lista pequeña: 9.300070814788342e-06 segundos
[SELECCION] Tiempo para una lista grande: 0.000213399951183796 segundos

[INSERCCION] Tiempo para lista pequeña: 6.8999361246824265e-06 segundos
[INSERCCION] Tiempo para lista grande: 0.00016410008538514376 segundos

[QUICKSORT] Tiempo para una lista pequeña: 1.8099904991686344e-05 segundos
[QUICKSORT] Tiempo para una lista grande: 0.00010679999832063913 segundos

[QUICKSORT ITER] Tiempo para una lista pequeña: 1.4099990949034691e-05 segundos
[QUICKSORT ITER] Tiempo para una lista grande: 6.740004755556583e-05 segundos

Lista pequeña ordenada por método BURBUJA
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPad>
```

Para 1000 elementos

```
ordenamiento.py U X
ordenamiento.py > ...
1 import random
2 import timeit
3 from generador_lista import generar_lista_desde_txt
4
5 # Generar una lista aleatoria de tamaño 1000
6 #archivo="personas1.txt"
7 #data=generar_lista_desde_txt(archivo)
8 #random.shuffle(data)
9 data=[random.randint(1, 1000000000) for _ in range(1000)]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPad> ^C
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPad>
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPad> cd 'c:\AMACION_TUPad'; & 'c:\Users\juana\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\juana\debugpy\launcher' '56057' '--' 'c:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_

[BURBUJA] Tiempo de ejecución para la lista pequeña fue de: 2.459995448589325e-05 segundos
[BURBUJA] Tiempo de ejecución para lista grande: 0.06424560002051294 segundos

[SELECCION] Tiempo para una lista pequeña: 1.309998333454132e-05 segundos
[SELECCION] Tiempo para una lista grande: 0.023188799968920648 segundos

[INSERCCION] Tiempo para lista pequeña: 1.0400079190731049e-05 segundos
[INSERCCION] Tiempo para lista grande: 0.020169599913060665 segundos

[QUICKSORT] Tiempo para una lista pequeña: 3.0099996365606785e-05 segundos
[QUICKSORT] Tiempo para una lista grande: 0.0013840000610798597 segundos

[QUICKSORT ITER] Tiempo para una lista pequeña: 1.9299914129078388e-05 segundos
[QUICKSORT ITER] Tiempo para una lista grande: 0.0009000999853014946 segundos

Lista pequeña ordenada por método BURBUJA
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPad>
```

Para 10000 elementos

```
ordenamiento.py U X
ordenamiento.py > ...
1 import random
2 import timeit
3 from generador_lista import generar_lista_desde_txt
4
5 # Generar una lista aleatoria de tamaño 1000
6 #archivo="personas1.txt"
7 #data=generar_lista_desde_txt(archivo)
8 #random.shuffle(data)
9 data=[random.randint(1, 1000000000) for _ in range(10000)]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPaD> ^C
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPaD>
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPaD> c:;; cd 'c:\User
AMACION_TUPaD'; & 'c:\Users\juana\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\juana\.vs
s\debugpy\launcher' '56067' '--' 'c:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROG

[BURBUJA] Tiempo de ejecución para la lista pequeña fue de: 1.4699995517730713e-05 segundos
[BURBUJA] Tiempo de ejecución para lista grande: 4.435122700058855 segundos

[SELECCION] Tiempo para una lista pequeña: 1.2599979527294636e-05 segundos
[SELECCION] Tiempo para una lista grande: 2.0819591999752447 segundos

[INSERCIÓN] Tiempo para lista pequeña: 1.0499963536858559e-05 segundos
[INSERCIÓN] Tiempo para lista grande: 2.0493923000758514 segundos

[QUICKSORT] Tiempo para una lista pequeña: 2.559996210038662e-05 segundos
[QUICKSORT] Tiempo para una lista grande: 0.016367799951694906 segundos

[QUICKSORT ITER] Tiempo para una lista pequeña: 1.8400023691356182e-05 segundos
[QUICKSORT ITER] Tiempo para una lista grande: 0.012533799977973104 segundos

Lista pequeña ordenada por método BURBUJA
PS C:\Users\juana\Downloads\TECNICATURA\A1 - PROGRAMACION I\TP_INTEG_PROGRAMACION_TUPaD>
```