Eficiencia de algoritmos

DGIIM 2017 - Grupo 2

Jesús Sánchez de Lechina Tejada
Álvaro López Jiménez
Miguel Ángel Robles Urquiza
Antonio Martín Ruiz

Índice

Eficiencia de algoritmos

1. Información de CPU

Nº de procesadores	4	Nº de procesadores	4
Procesadores	Intel Core i5-5200U CPU @ 2.20GHz	Procesadores	Intel Core i7-4500U CPU @ 1.80GHz
Tamaño de caché	3072 KB	Tamaño de caché	4096 KB
Memoria	~ 16GB	Memoria	~ 8GB
Sistema Operativo	Ubuntu 14.04.5 x86_64	Sistema Operativo	Ubuntu 16.10 LTS 64-bits
Nº de procesadores	4	Nº de procesadores	4
Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz	Procesadores	Intel Core i5-5200U CPU @ 2.20GHz
Tamaño de caché	3863 KB	Tamaño de caché	3072 KB
Memoria	~ 4GB	Memoria	~ 8GB
Sistema Operativo	Ubuntu 16.04.2 LTS 64-bits	Sistema Operativo	Ubuntu 16.04.2 LTS 64-bits

2.1 Eficiencia $O(n^2)$

- 2.1.1 Burbuja
- 2.1.2 Inserción
- 2.1.3 Selección
- 2.1.4 Comparación

2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz

- Ordena un vector de tamaño n.
- Tiene una eficiencia de $O(n^2)$ en el peor de los casos.
- Consiste en ciclar repetidamente a través de la lista, comparando elementos adyacentes de dos en dos. Si un elemento es mayor que el que está en la siguiente posición se intercambian.

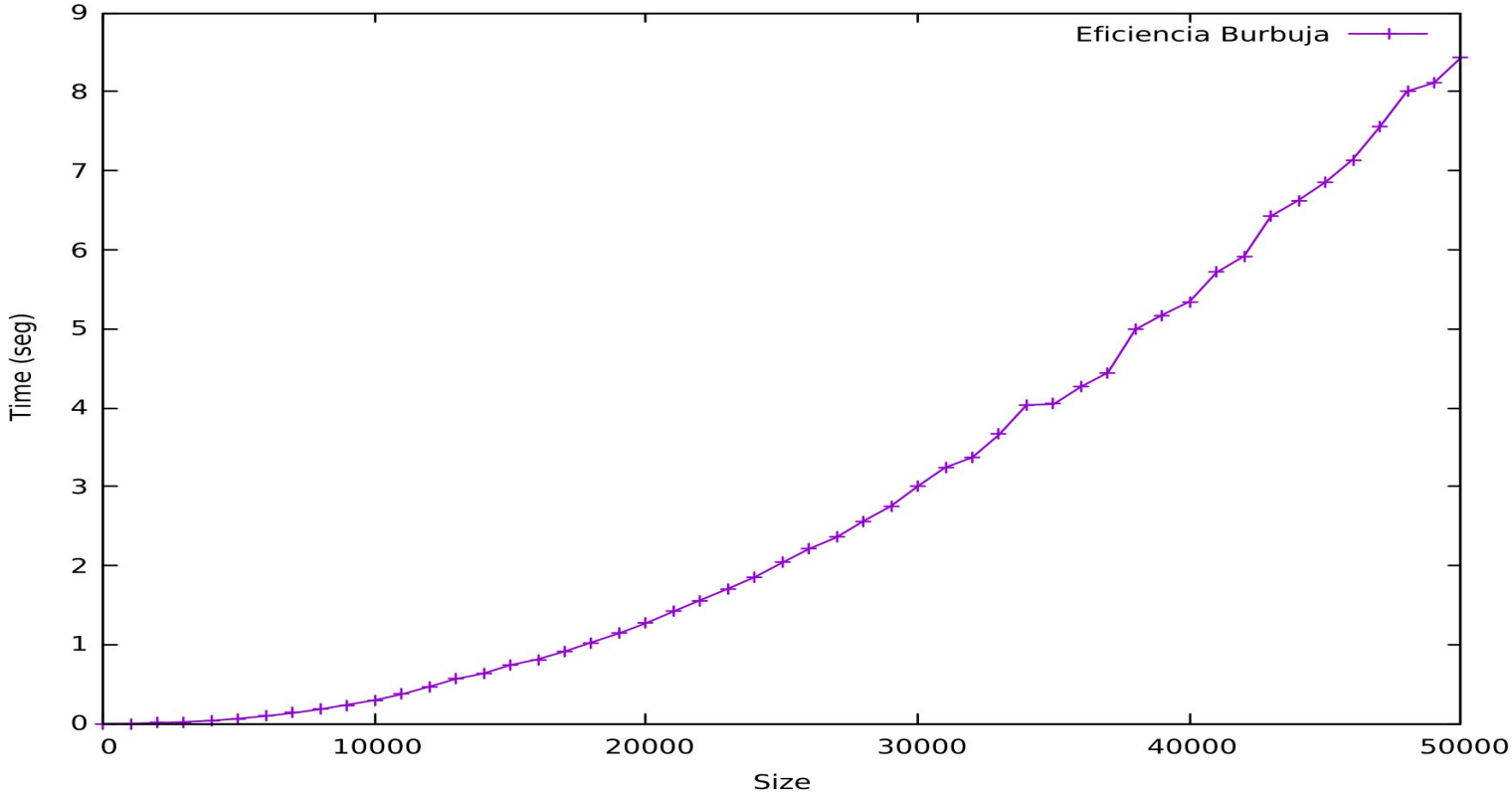
2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.041315
15000	0.346635
25000	0.967005
35000	1.90573
45000	3.11329

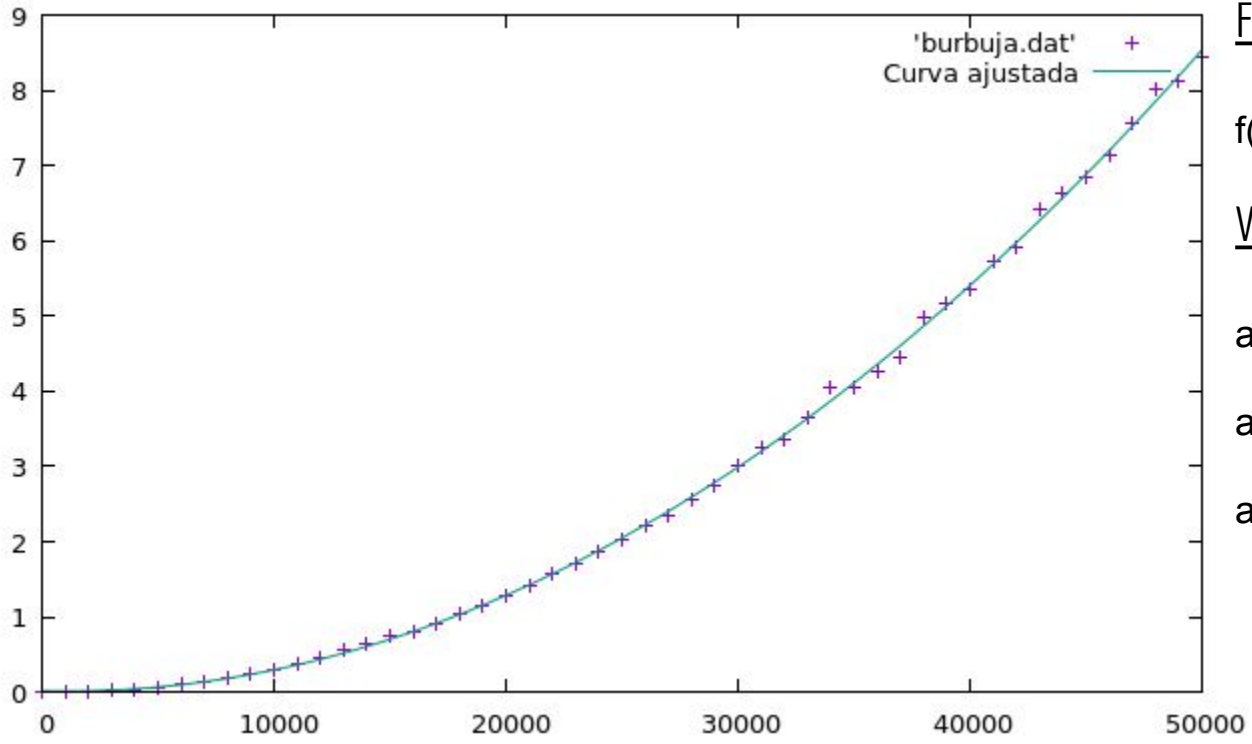
2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



FUNCIÓN

f(x) = a0*x*x+a1*x+a2

VARIABLES OCULTAS

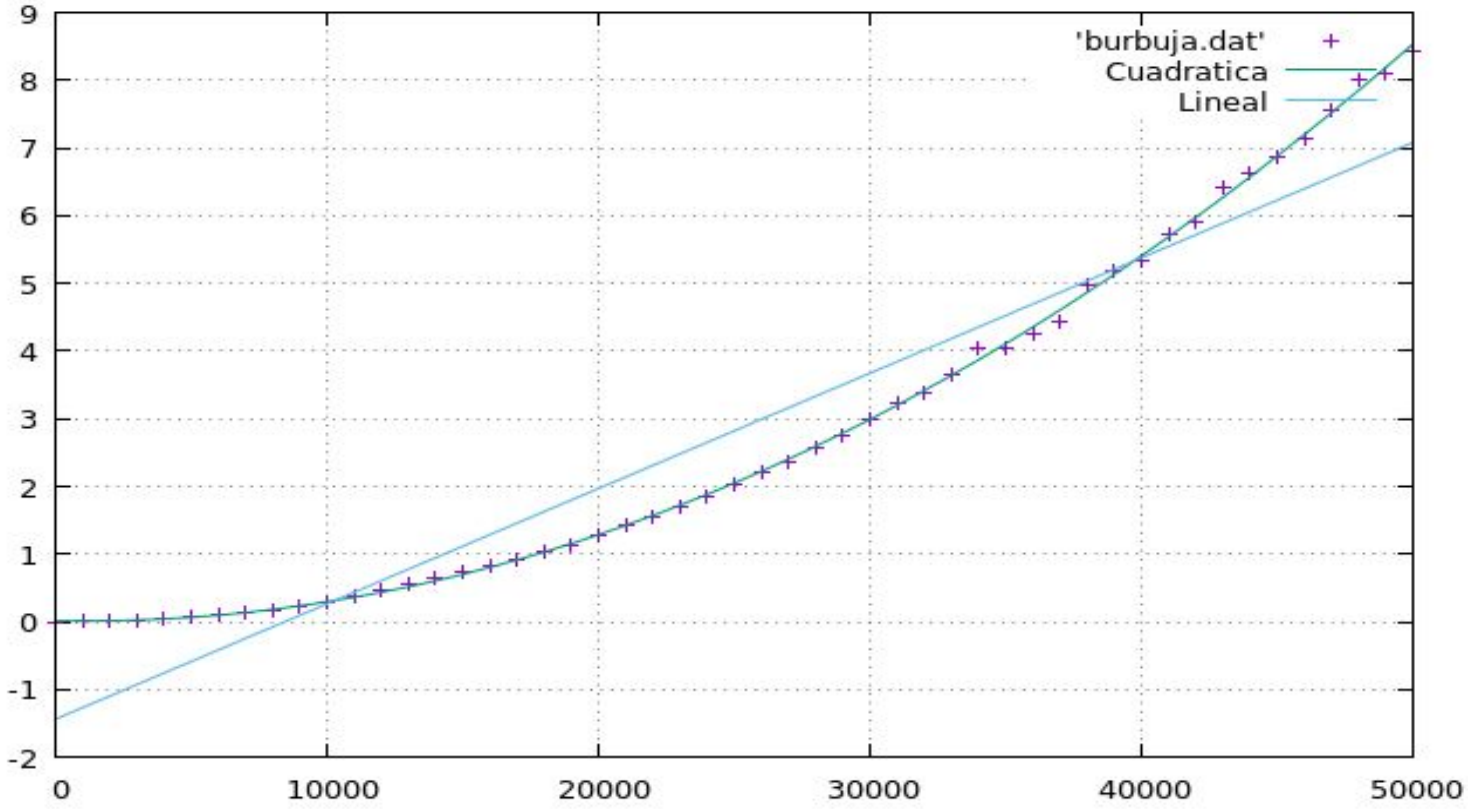
a0 = 1.29827e-09

a1 = -8.31901e-07

a2 = 0.00602881

2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

- Ordena un vector de tamaño n.
- Tiene una eficiencia de $O(n^2)$ en el peor de los casos.
- Consiste en dividir el vector en dos subvectores (ordenado y no ordenado). En un primer lugar, el vector ordenado sólo tendrá un elemento (el primero). Para cada elemento restante del vector se compara con la parte ordenada y se inserta en el lugar adecuado.

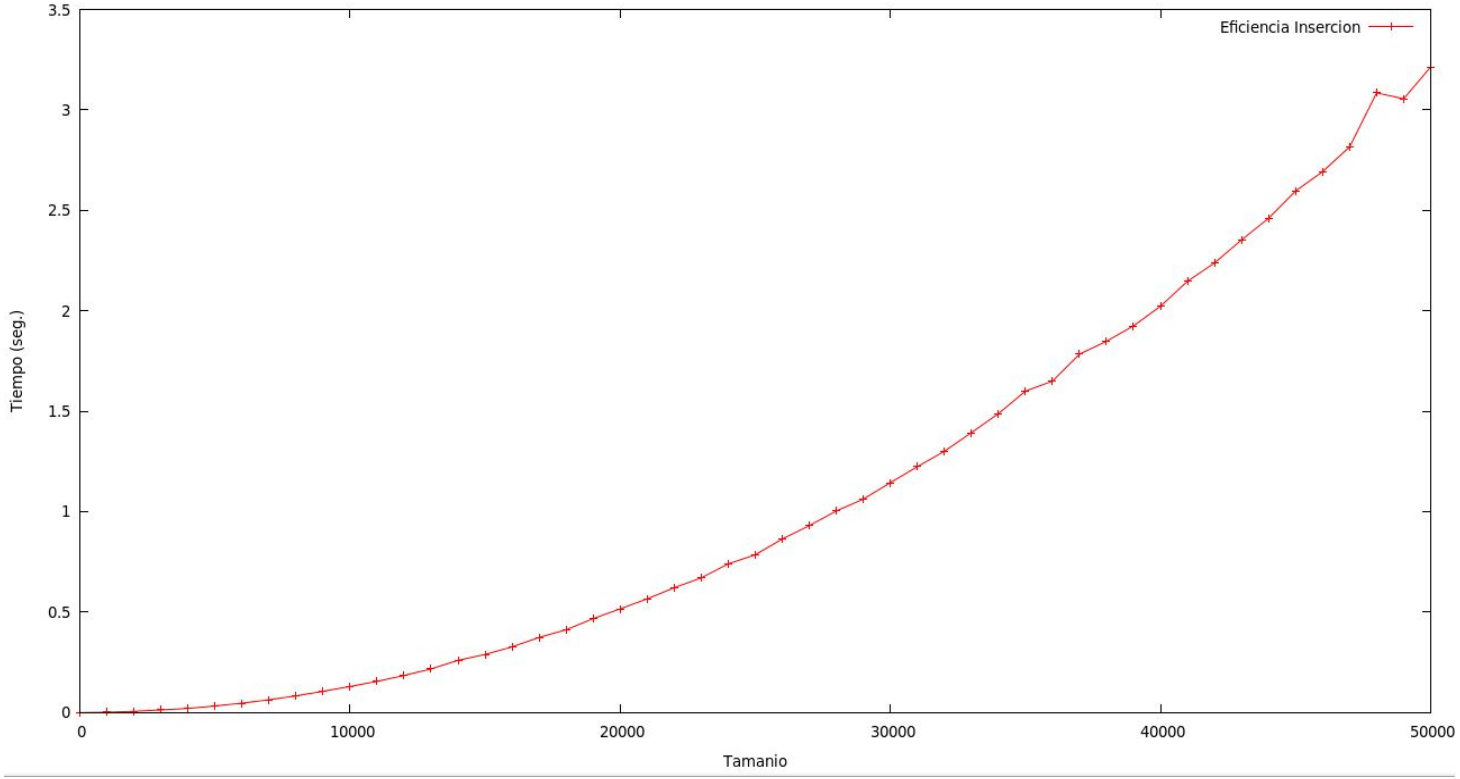
2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.032399
15000	0.289151
25000	0.784994
35000	1.59976
45000	2.59602

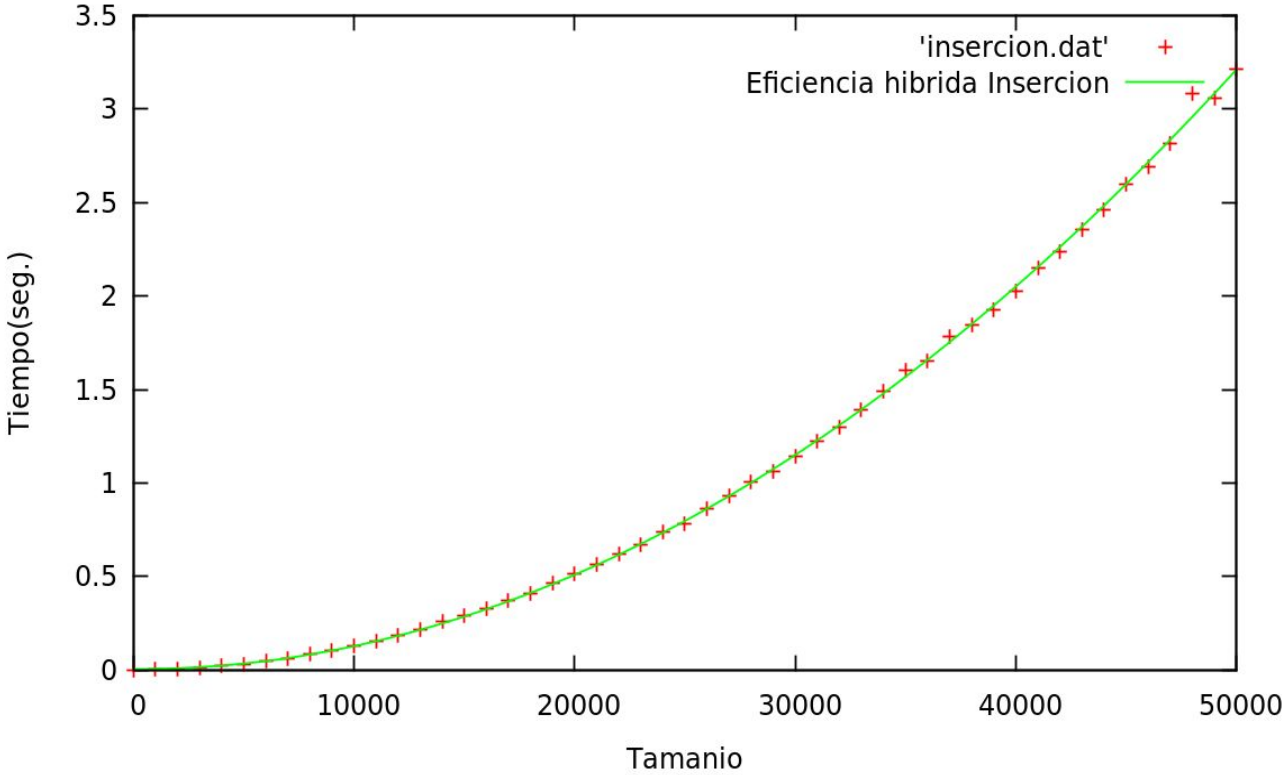
2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



FUNCIÓN

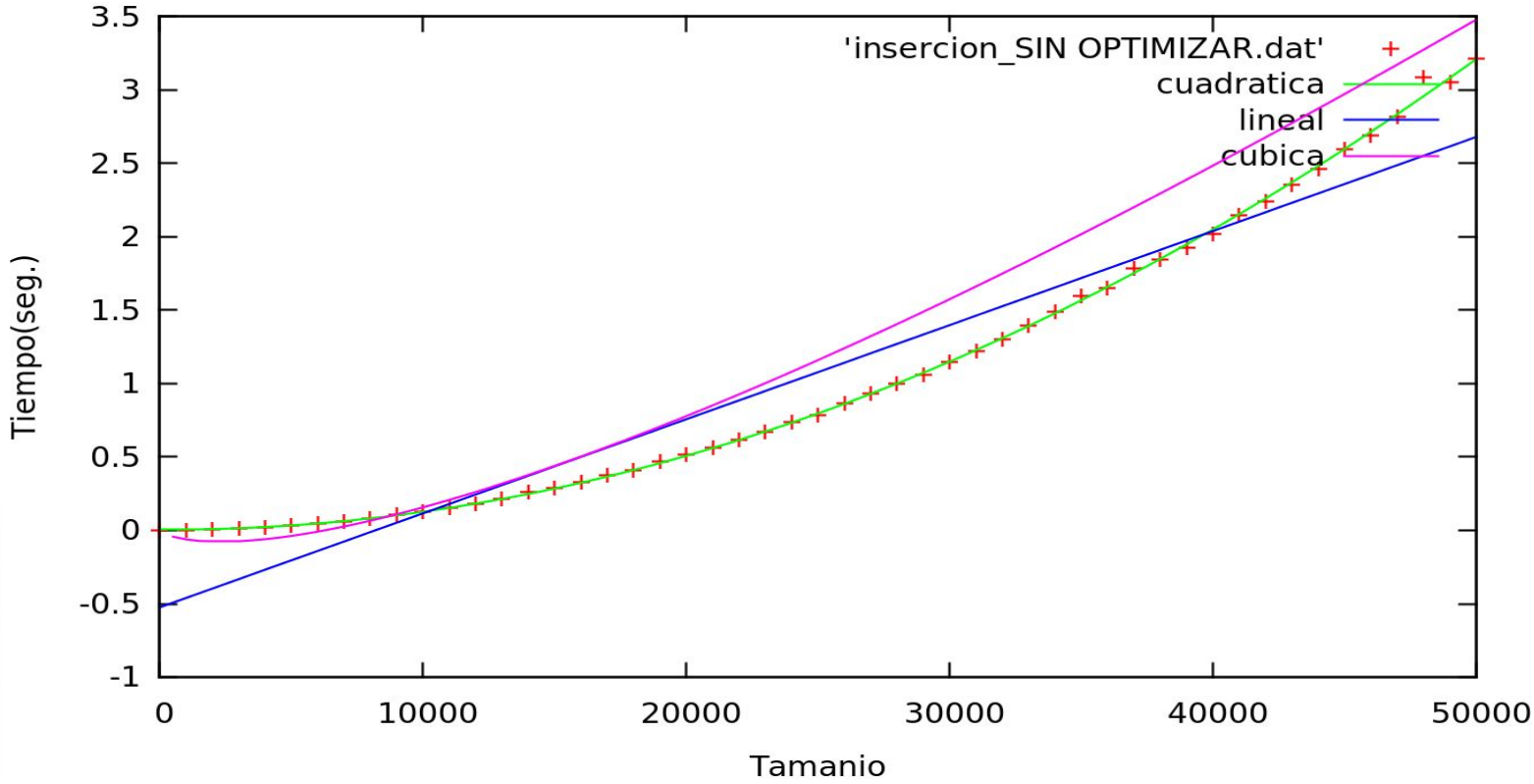
f(x) = a0*x*x+a1*x+a2

VARIABLES OCULTAS

a0 = 3.56201e-09
a1 = -7.73511e-06
a2 = 0.0164765

2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

- Ordena un vector de tamaño n
- Tiene una eficiencia de $O(n^2)$.
- Busca el mínimo elemento entre la posición i y el final y lo coloca en la posición i, con i = 1, 2, ... , n. Por lo tanto el proceso se realiza en n iteraciones.

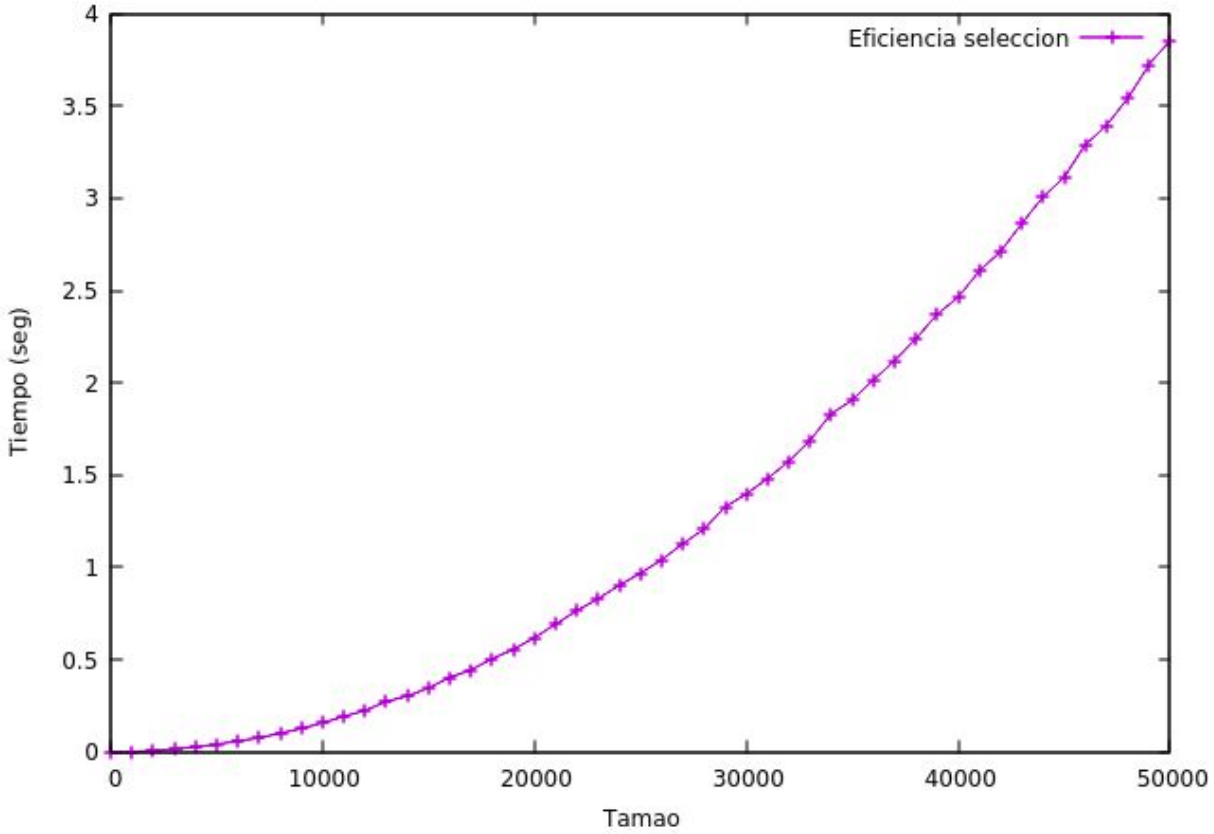
2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.041315
15000	0.346635
25000	0.967005
35000	1.90573
45000	3.11329

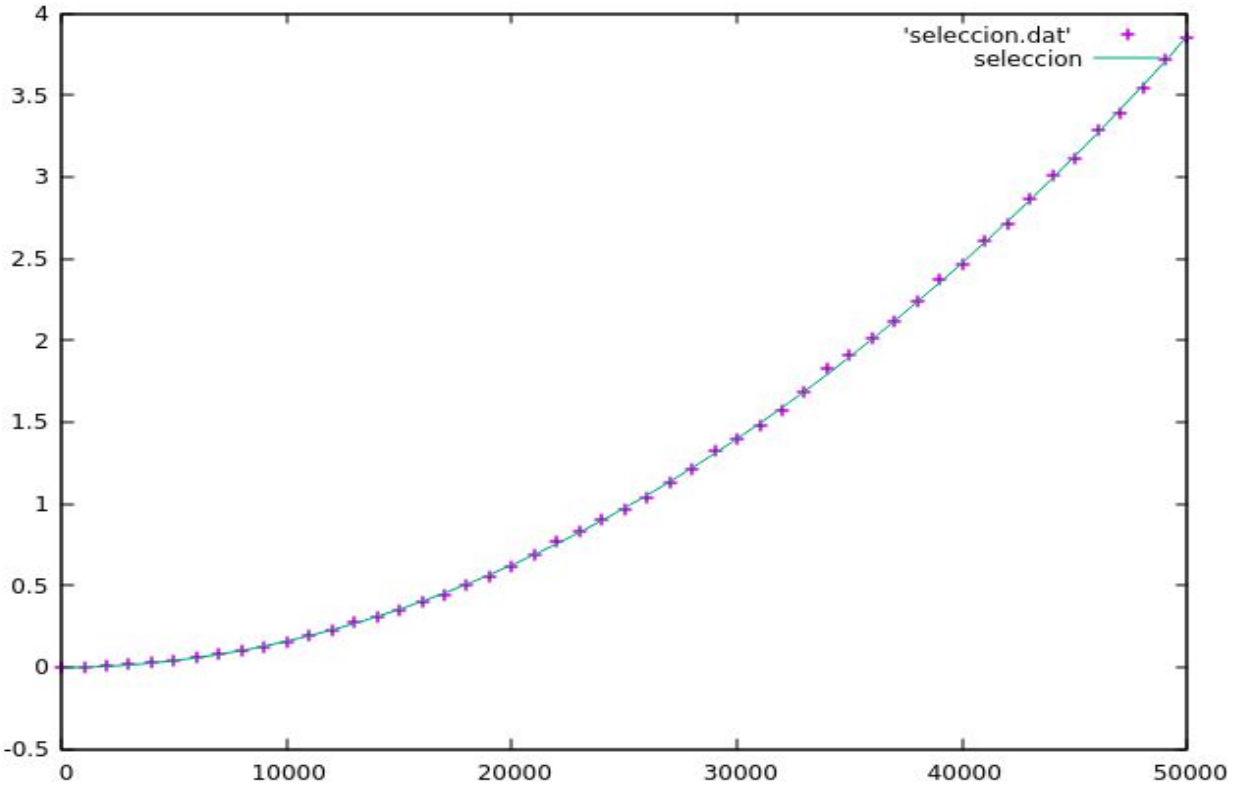
2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



FUNCIÓN

f(x) = a0*x*x+a1*x+a2

VARIABLES OCULTAS

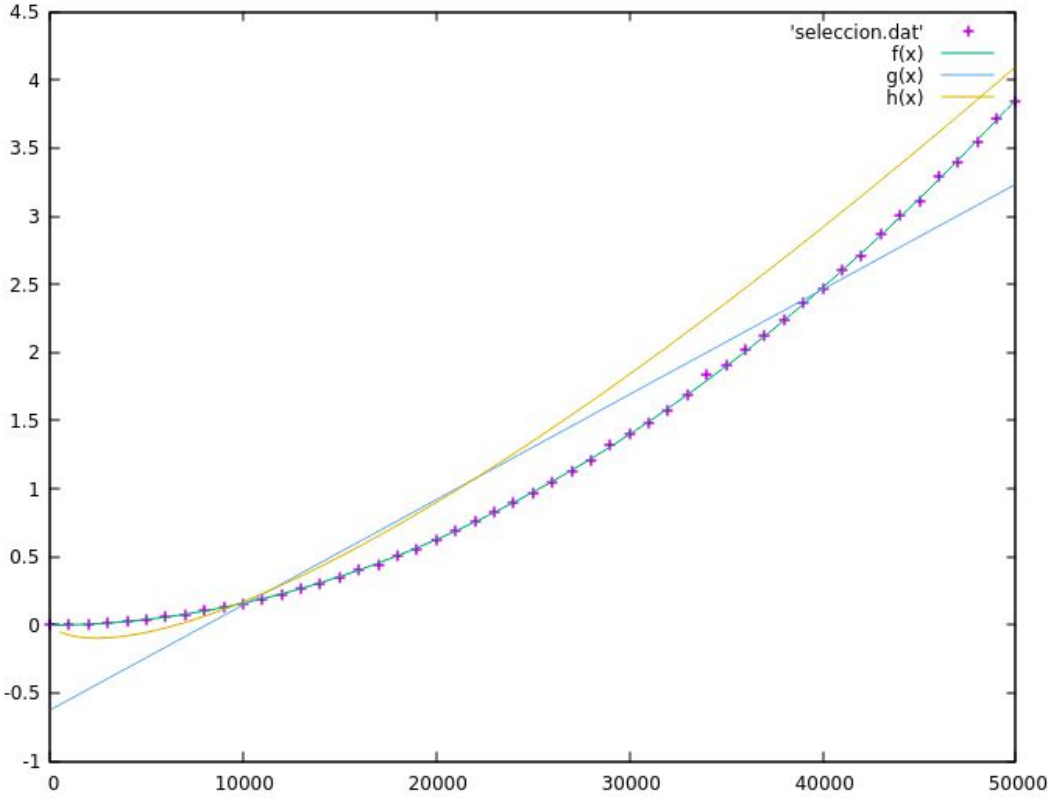
a0 = 1.52682e-09

a1 = 8.89438e-07

a2 = -0.00336106

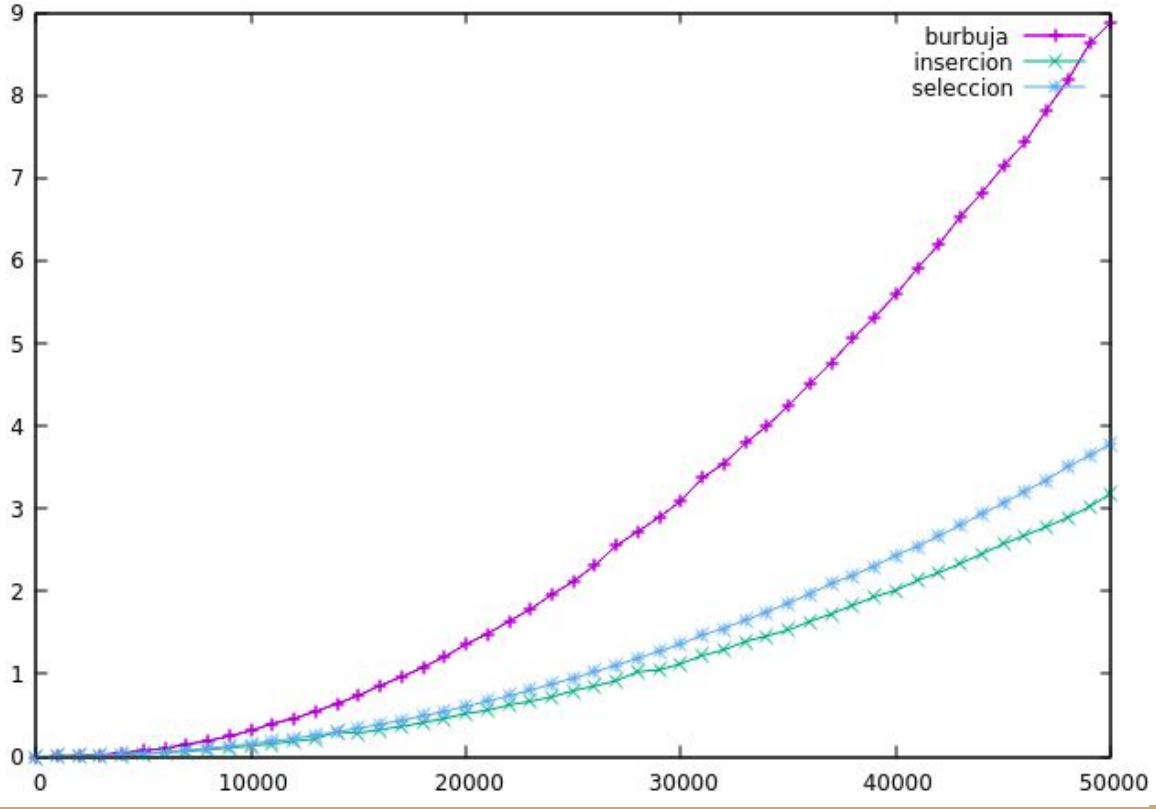
2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.1.4 Comparación

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.2 Eficiencia O(nlogn)

- 2.2.1 Mergesort
- 2.2.2 Quicksort
- 2.2.3 Heapsort
- 2.2.4 Comparación

2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz

- Ordena un vector de tamaño n.
- Tiene una eficiencia de O(nlogn)
- Si el vector es vacío o de tamaño uno está ordenado. En caso contrario divide en vectores de la mitad del tamaño que el original, se dividen las sublistas recursivamente (usando este procedimiento), y posteriormente se mezclan estas dos sublistas en una sola lista ordenada.

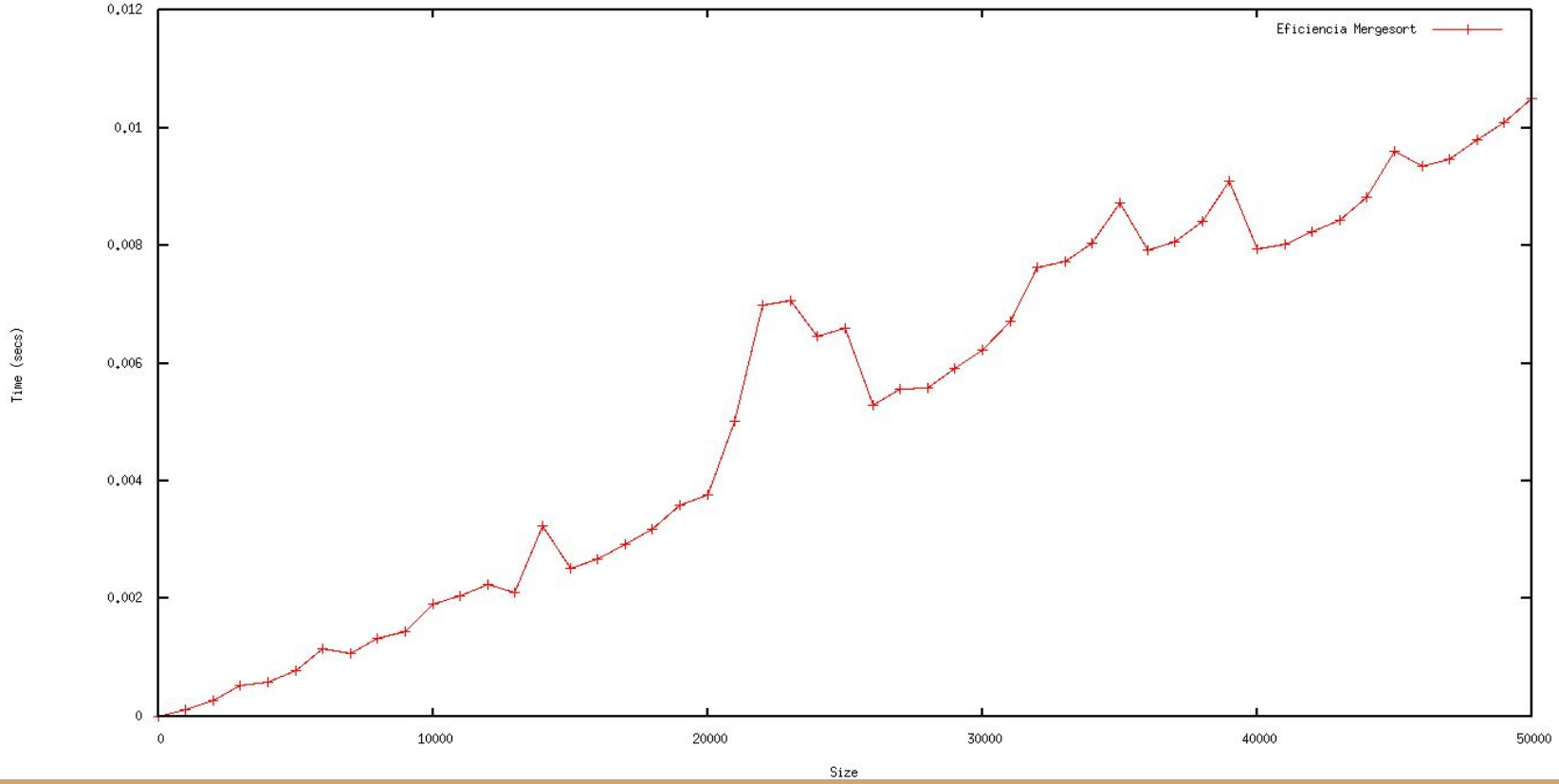
2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.000782538
15000	0.002519
25000	0.00659
35000	0.008736
45000	0.009597

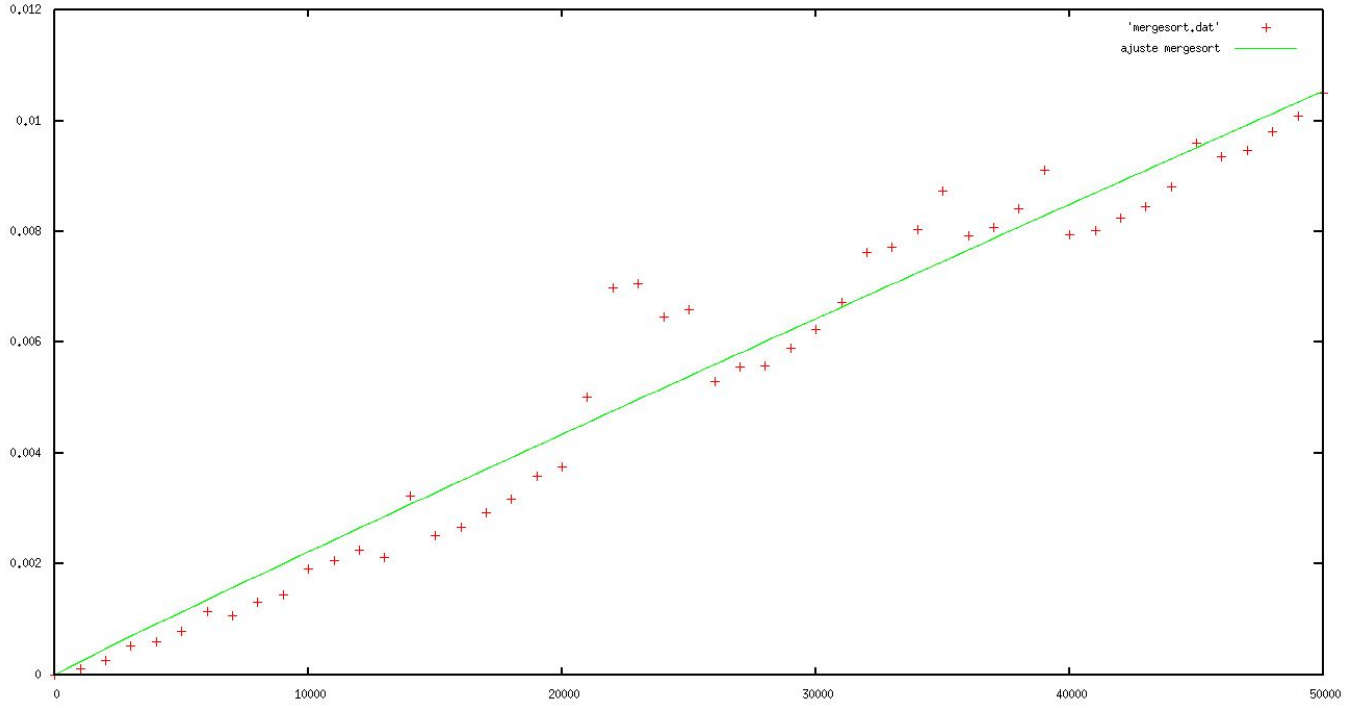
2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



FUNCIÓN

f(x) = a0*x*x+a1*x+a2

VARIABLES OCULTAS

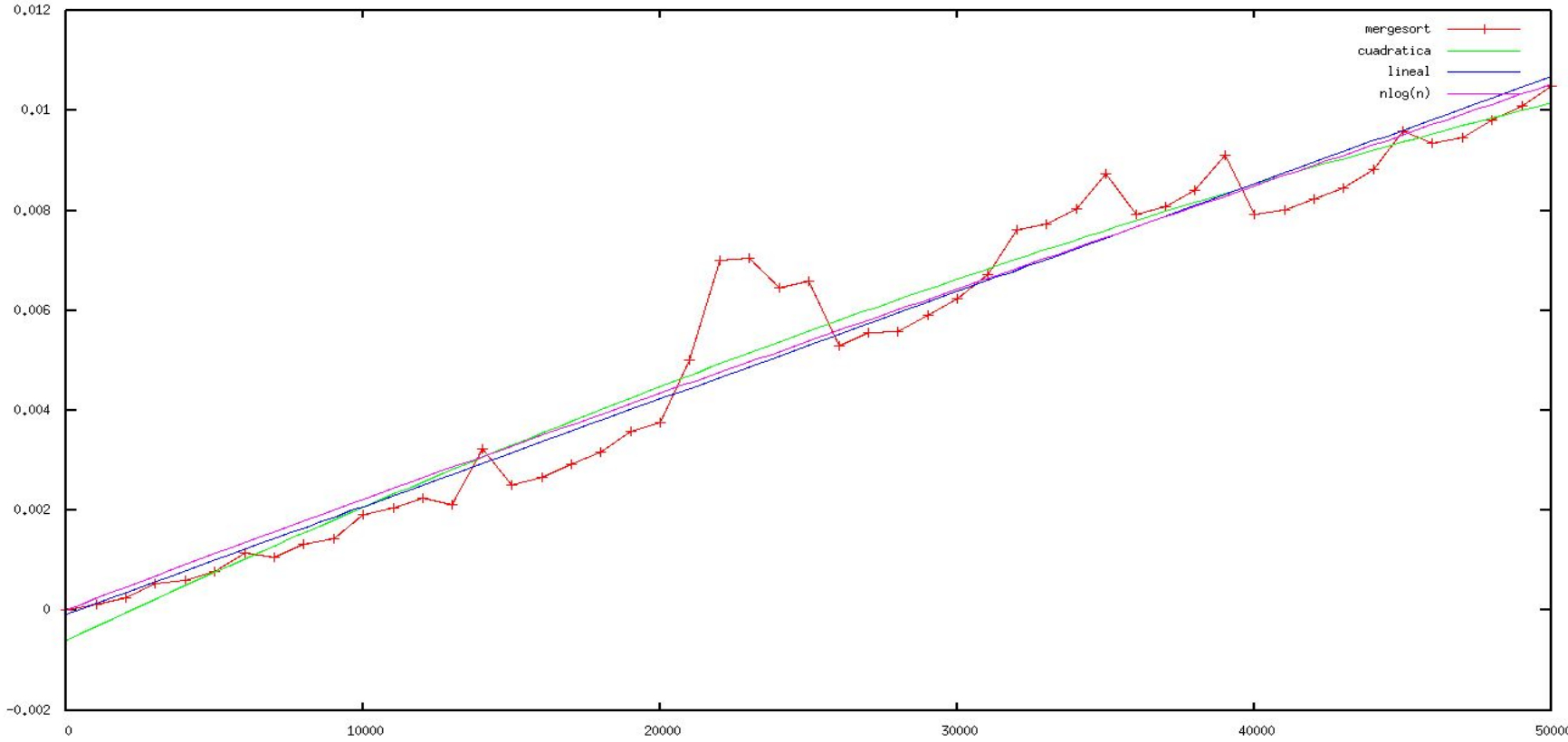
a0 = -0.253707

a1 = 2.74939e-08

a2 = 0.253707

2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz

- Ordena un vector de tamaño n.
- Tiene una eficiencia de O(nlogn), aunque en el peor de los casos puede tener una eficiencia de $O(n^2)$, ya que el número de comparaciones no depende del orden de los términos, si no del número de términos
- Se elige un elemento v de la lista L de elementos al que se le llama pivote. Se particiona la lista L en tres listas:
  - L1 - que contiene todos los elementos de L menos v que sean menores o iguales que v
  - L2 - que contiene a v
  - L3 - que contiene todos los elementos de L menos v que sean mayores o iguales que v
- Se aplica la recursión sobre L1 y L3 y se concatenan todas al final

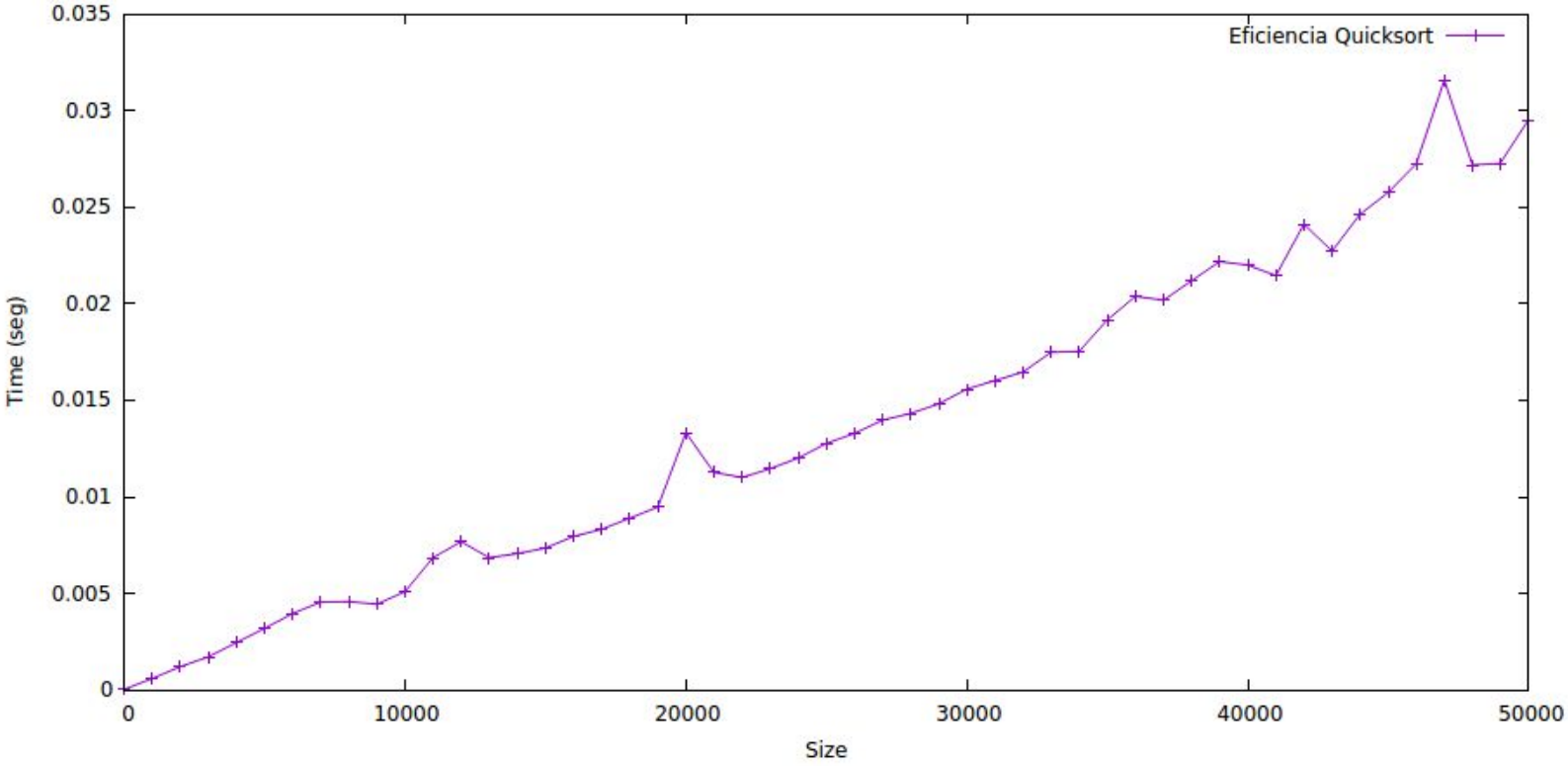
2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.003185
15000	0.007348
25000	0.012763
35000	0.019164
45000	0.025764

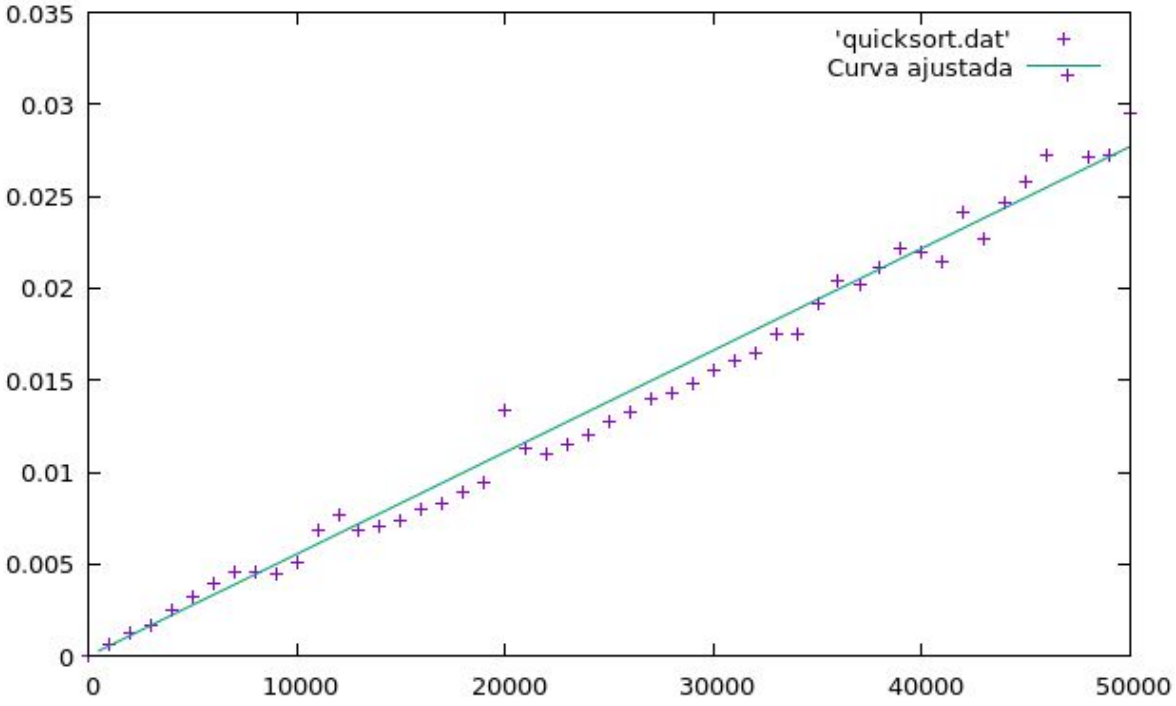
2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



FUNCIÓN

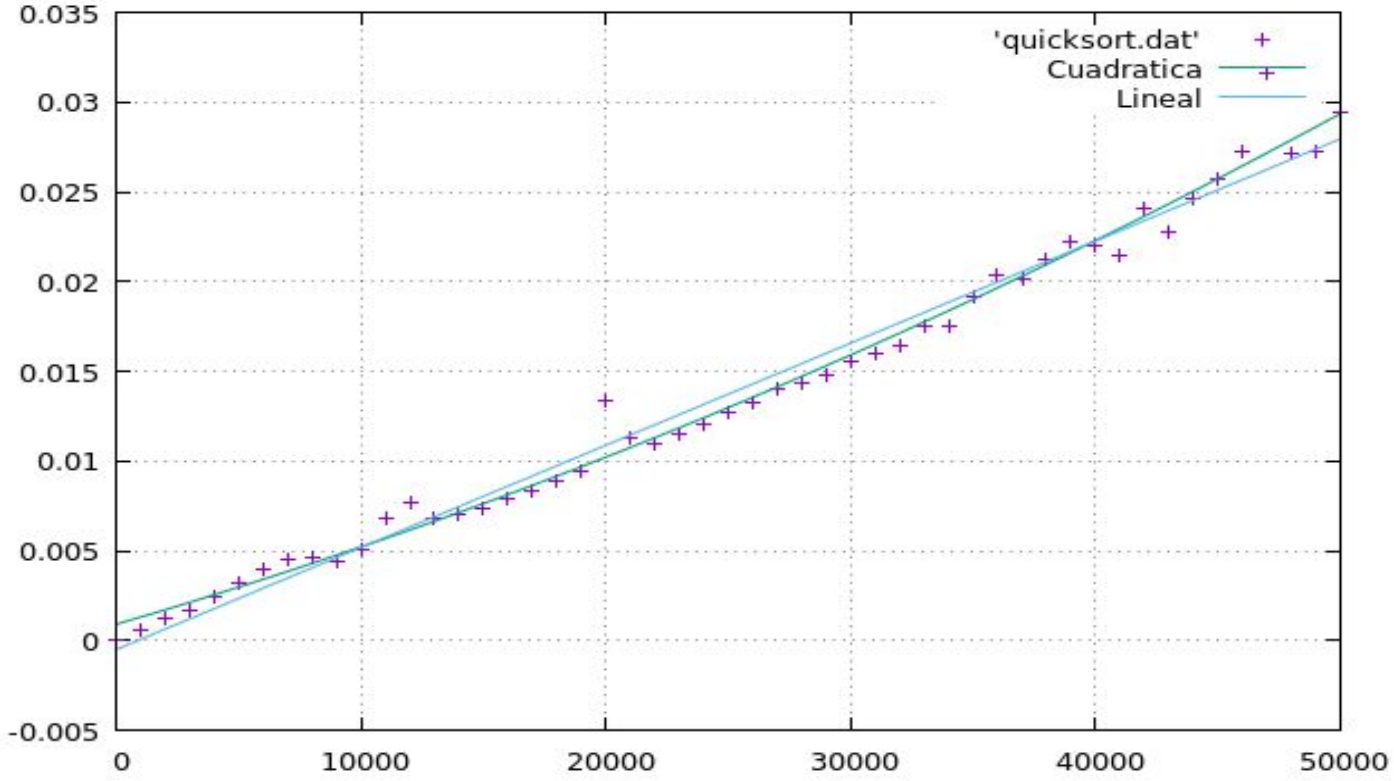
f(x) = x*a0*x**a1+a2*x

VARIABLES OCULTAS

a0 = 3.56201e-09
a1 = -7.73506e-06
a2 = 5.50681e-07

2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

- Ordena un vector de tamaño n.
- Tiene una eficiencia de O(nlogn).
- Consiste en almacenar todos los elementos del vector a ordenar en un montículo (*heap*), y luego extraer el nodo que queda como nodo raíz del montículo (cima) en sucesivas iteraciones obteniendo el conjunto ordenado. Basa su funcionamiento en una propiedad de los montículos, por la cual, la cima contiene siempre el menor elemento (o el mayor, según se haya definido el montículo) de todos los almacenados.

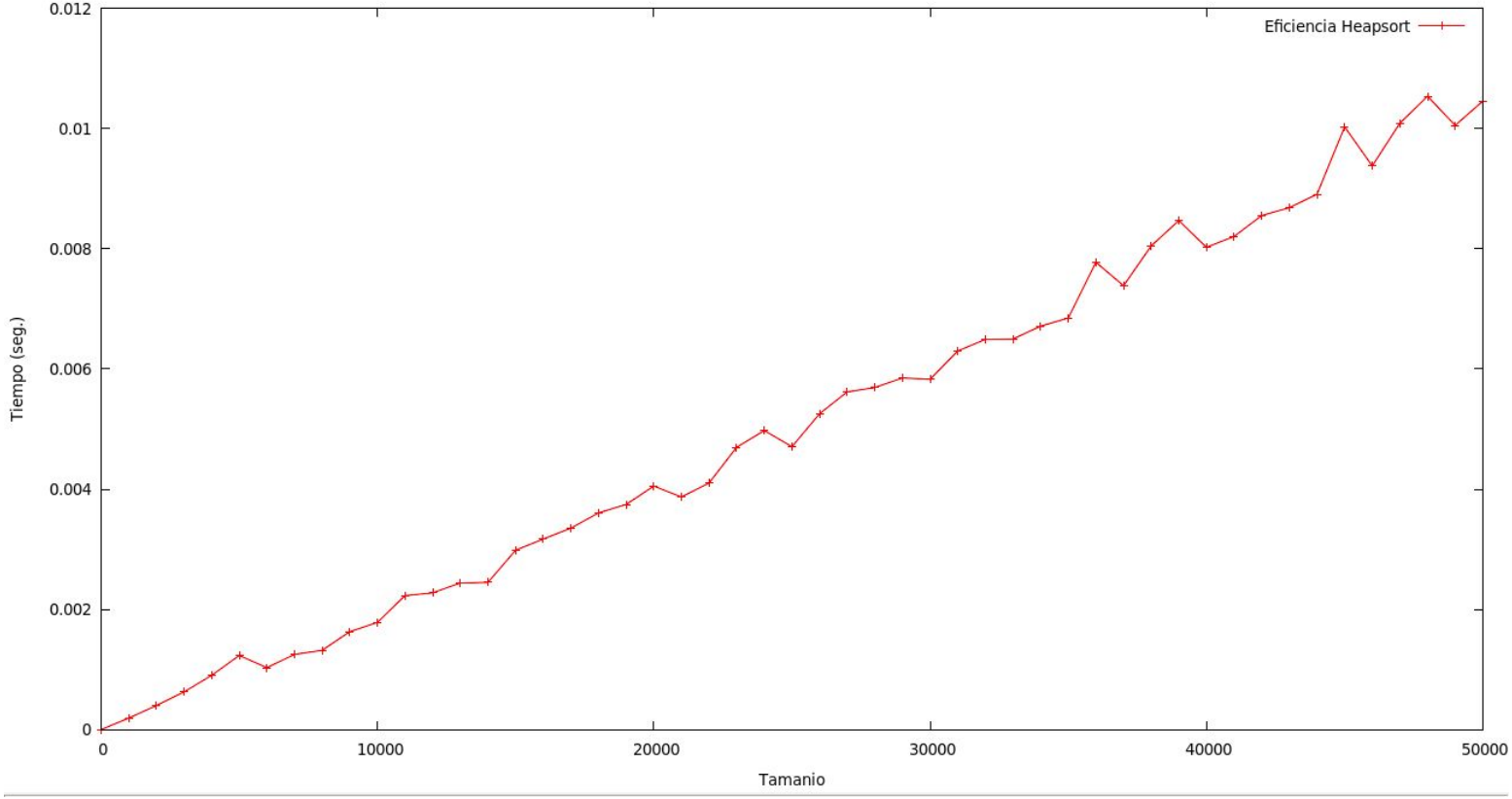
2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.001234
15000	0.002985
25000	0.004707
35000	0.006847
45000	0.010026

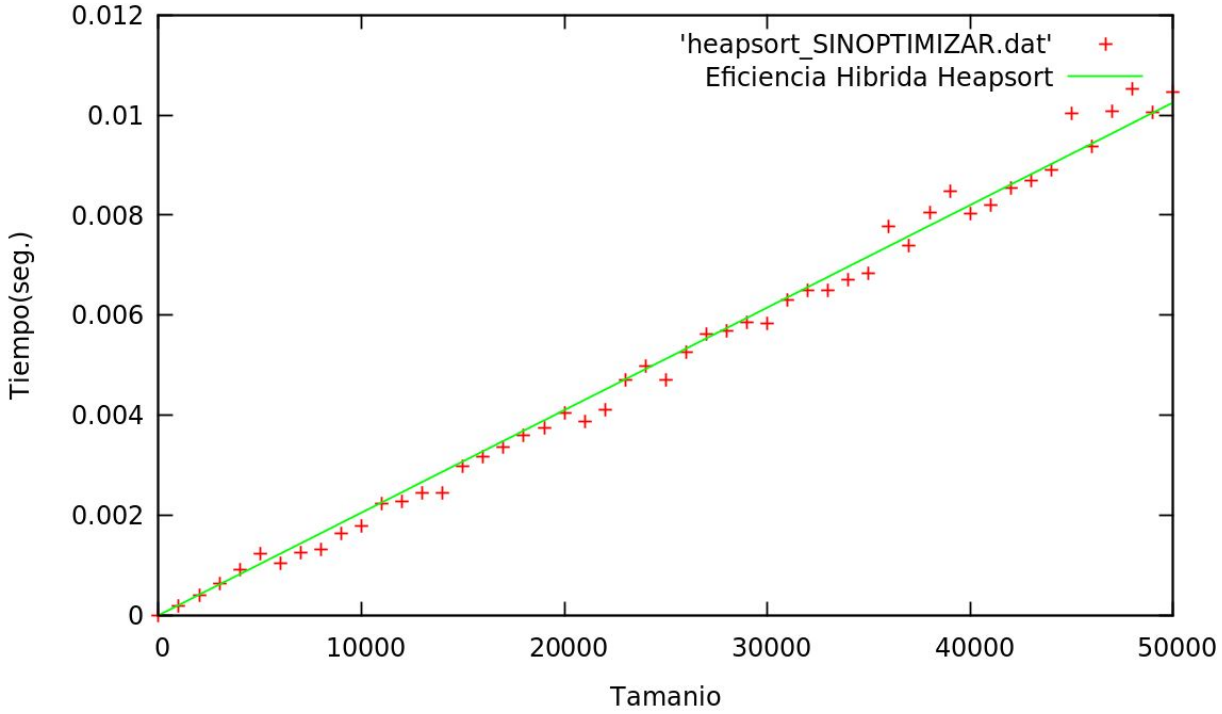
2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



FUNCIÓN

f(x) = x*a0*x**a1+a2*x

VARIABLES OCULTAS

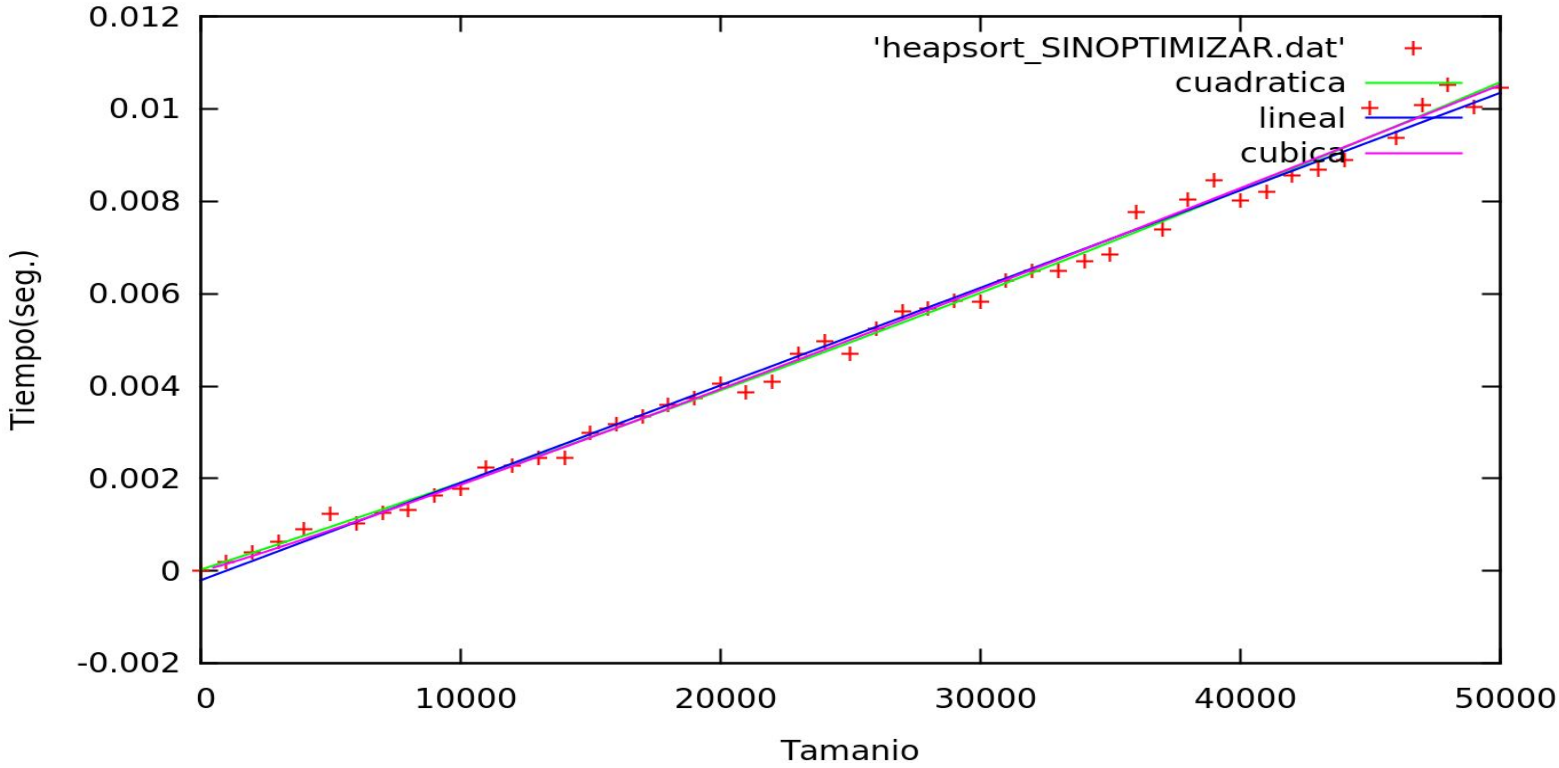
a0 = 1.95648e-05

a1 = 0.0661408

a2 = 1.95648e-05

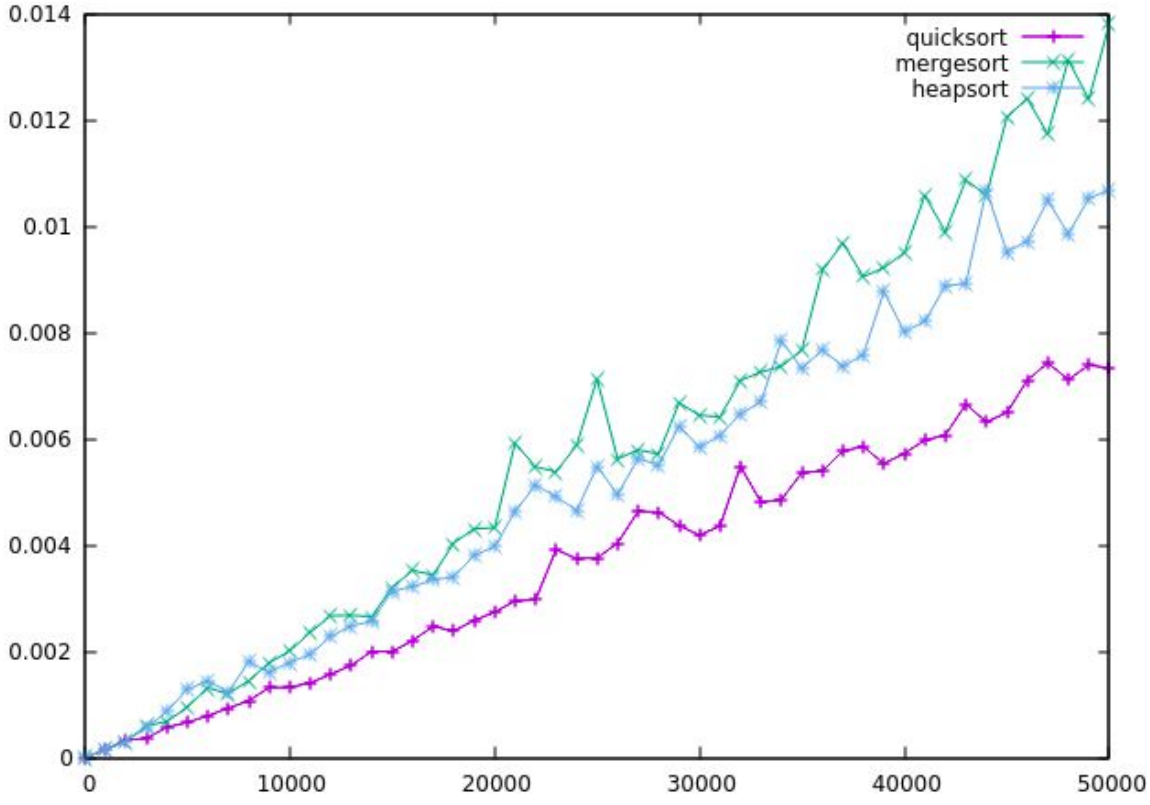
2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



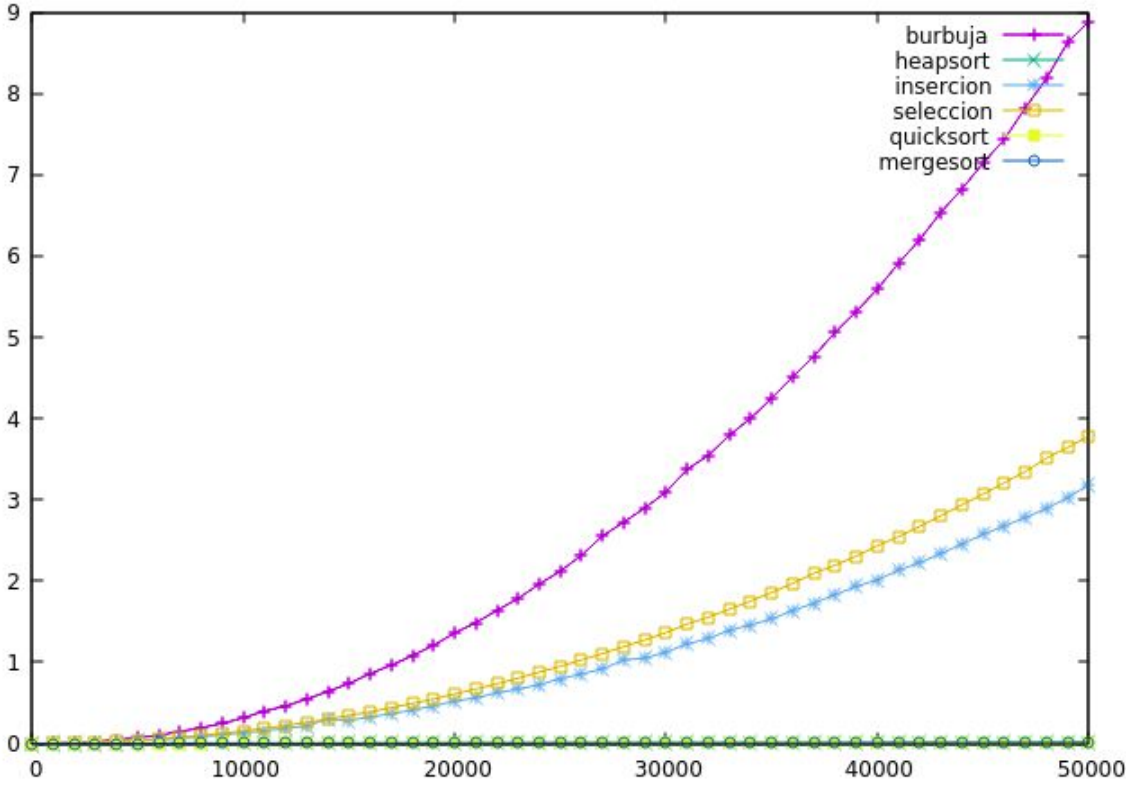
2.2.4 Comparación

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.3 Comparación

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



3. Algoritmo de Floyd
4. Algoritmo de Hanoi

3. Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

- Encuentra caminos mínimos entre grafos dirigidos ponderados.
- Tiene una eficiencia de $O(n^3)$.
- El algoritmo compara todos los posibles caminos entre cada par de nodos del grafo y selecciona el mínimo. En cada iteración almacena los resultados en una matriz. Por tanto, en una sola ejecución se obtienen todos los caminos mínimos entre cada par de nodos del grafo.

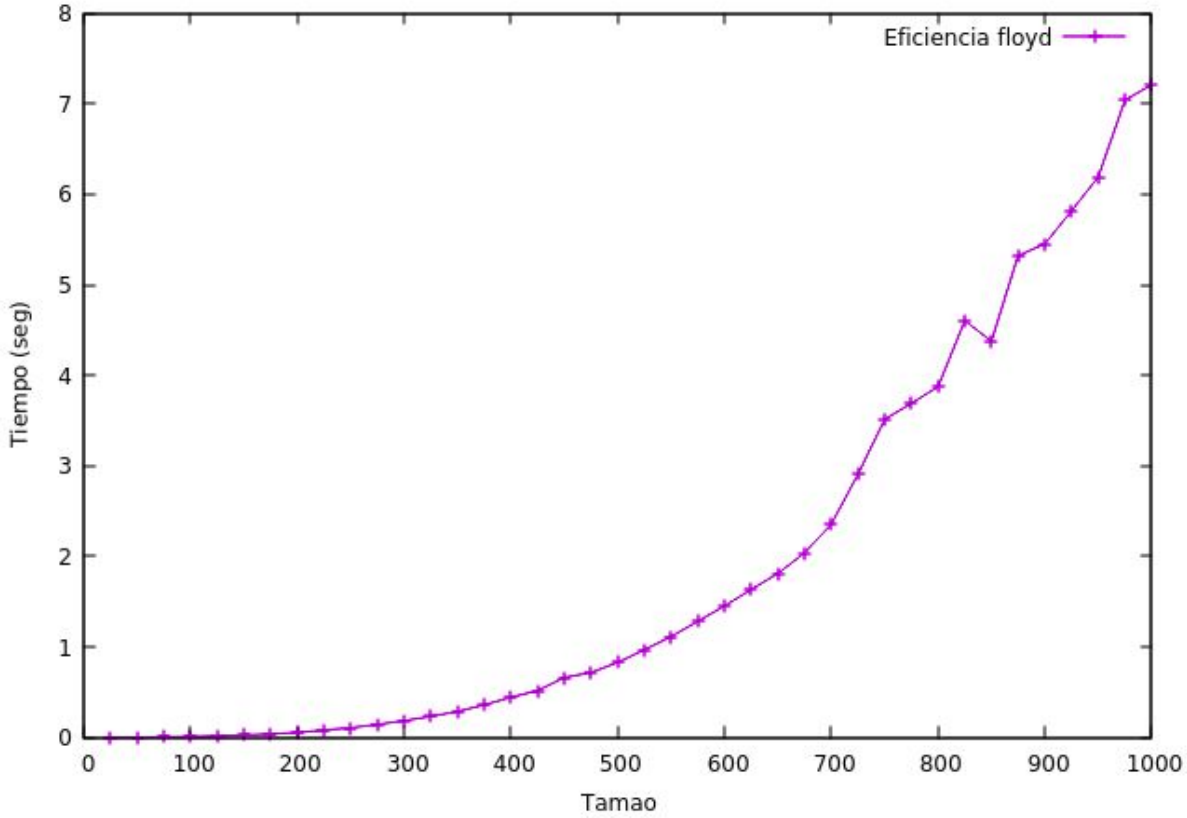
3. Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.041315
15000	0.346635
25000	0.967005
35000	1.90573
45000	3.11329

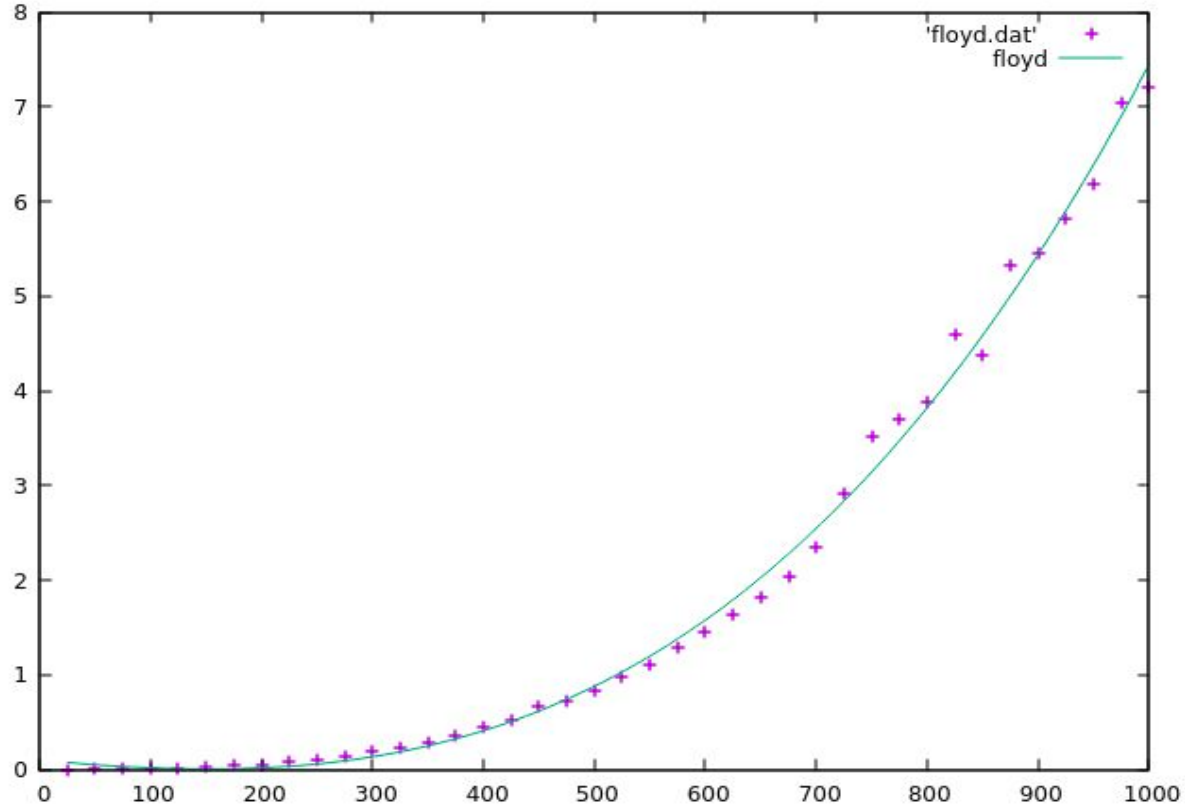
3. Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



3. Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



FUNCIÓN

f(x) = a0*x*x*x+a1*x*x+a2*x+a3

VARIABLES OCULTAS

a0 = 6.24086e-09

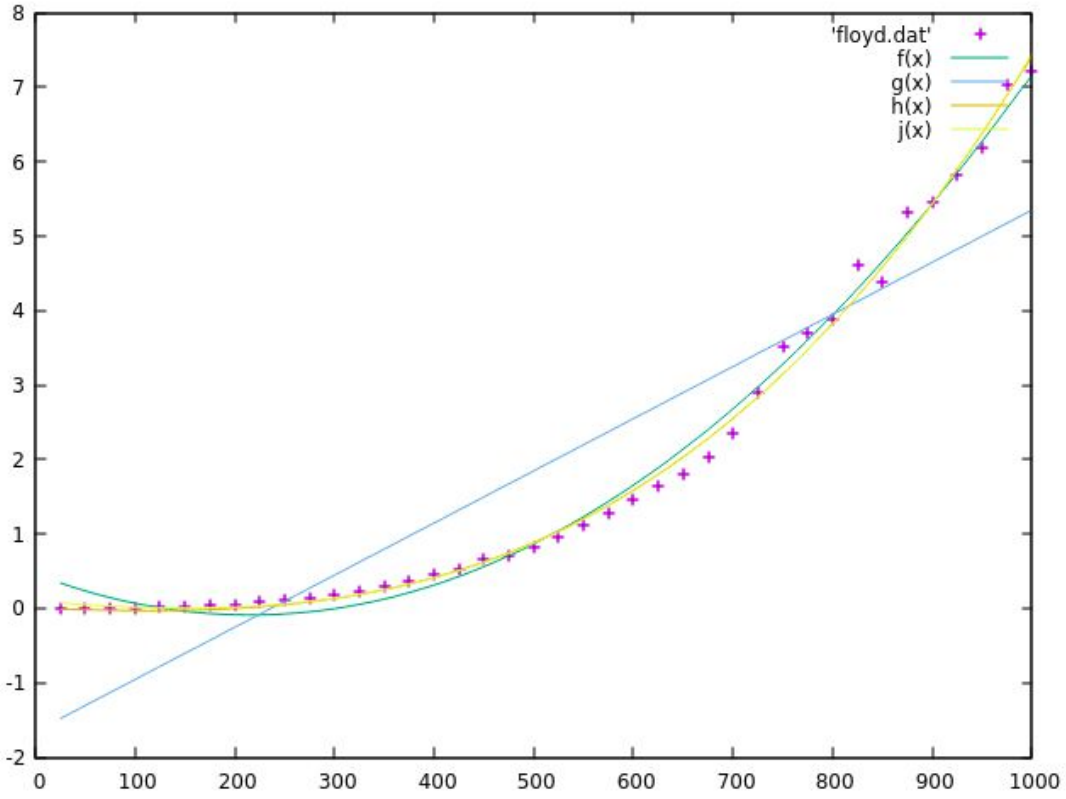
a1 = 2.20775e-06

a2 = -0.00110777

a3 = 0.0990287

3. Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



2.1.3 Hanoi

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz

- Resuelve el problema matemático de las torres de Hanoi
- Tiene una eficiencia de $O(2^n)$
- Consiste en la resolución recursiva de este algoritmo. Si el número de discos es 1, simplemente mueve disco a otra varilla y termina el problema. En caso contrario se llama recursivamente para desplazar los discos a su varilla correspondiente.

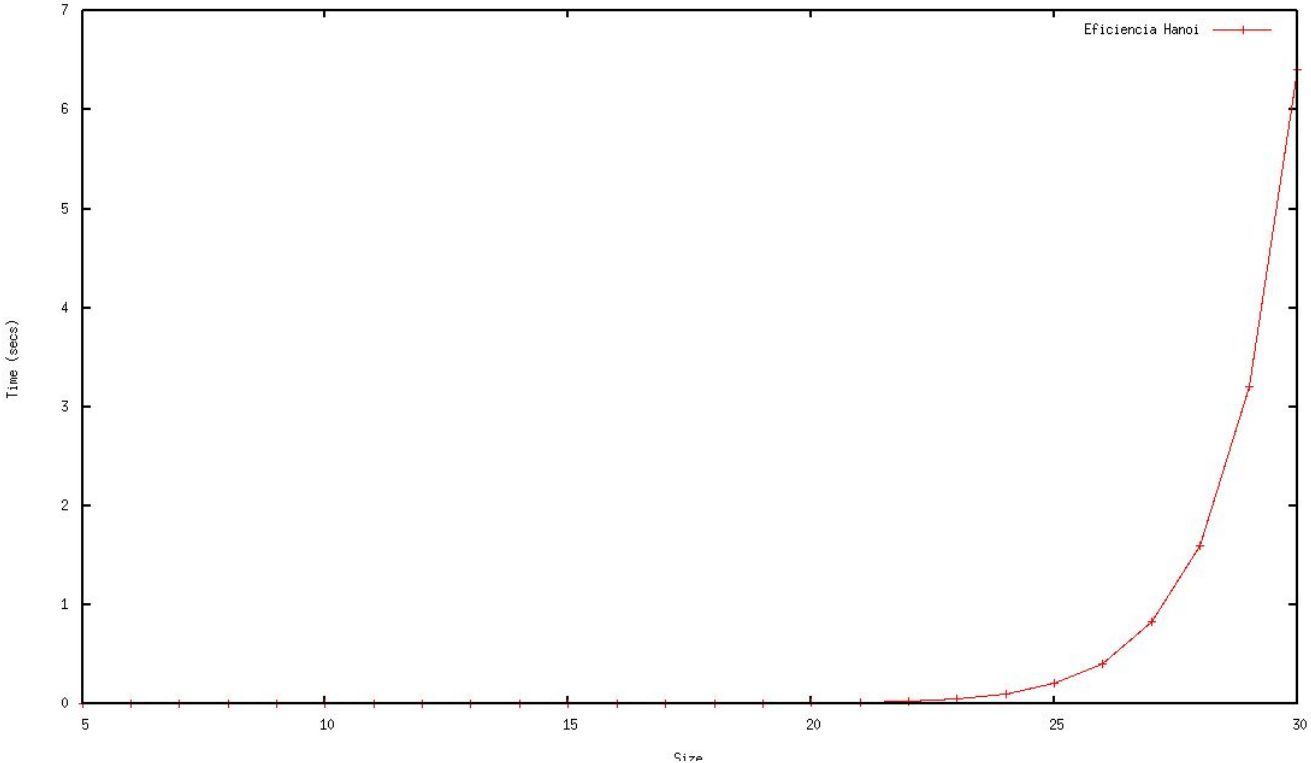
2.1.3 Hanoi

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz

TAMAÑO	TIEMPO (SEG.)
5000	0.000782538
15000	0.002519
25000	0.00659
35000	0.008736
45000	0.009597

2.1.3 Hanoi

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



FUNCIÓN

f(x) = 2**(ax+b)

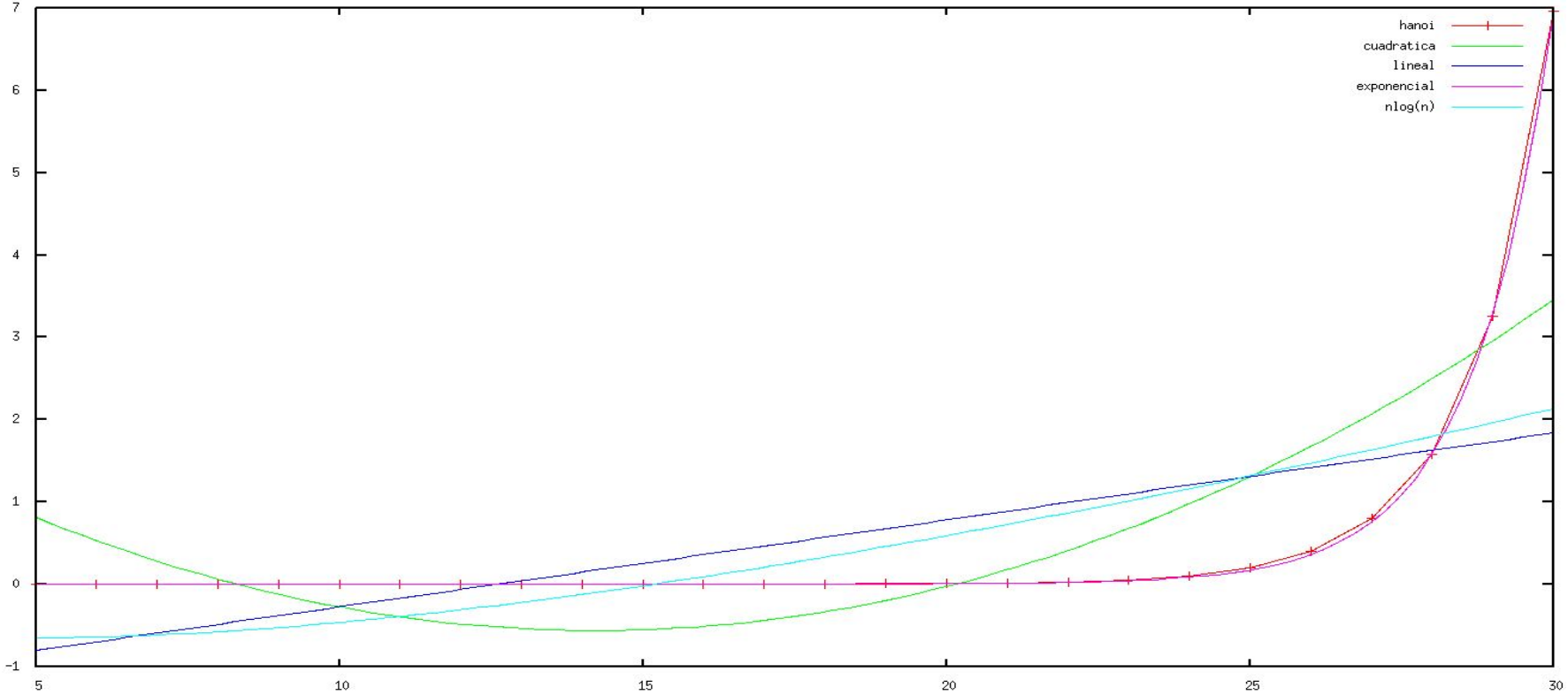
VARIABLES OCULTAS

a = 1.06798

b = -29.2424

2.1.3 Hanoi

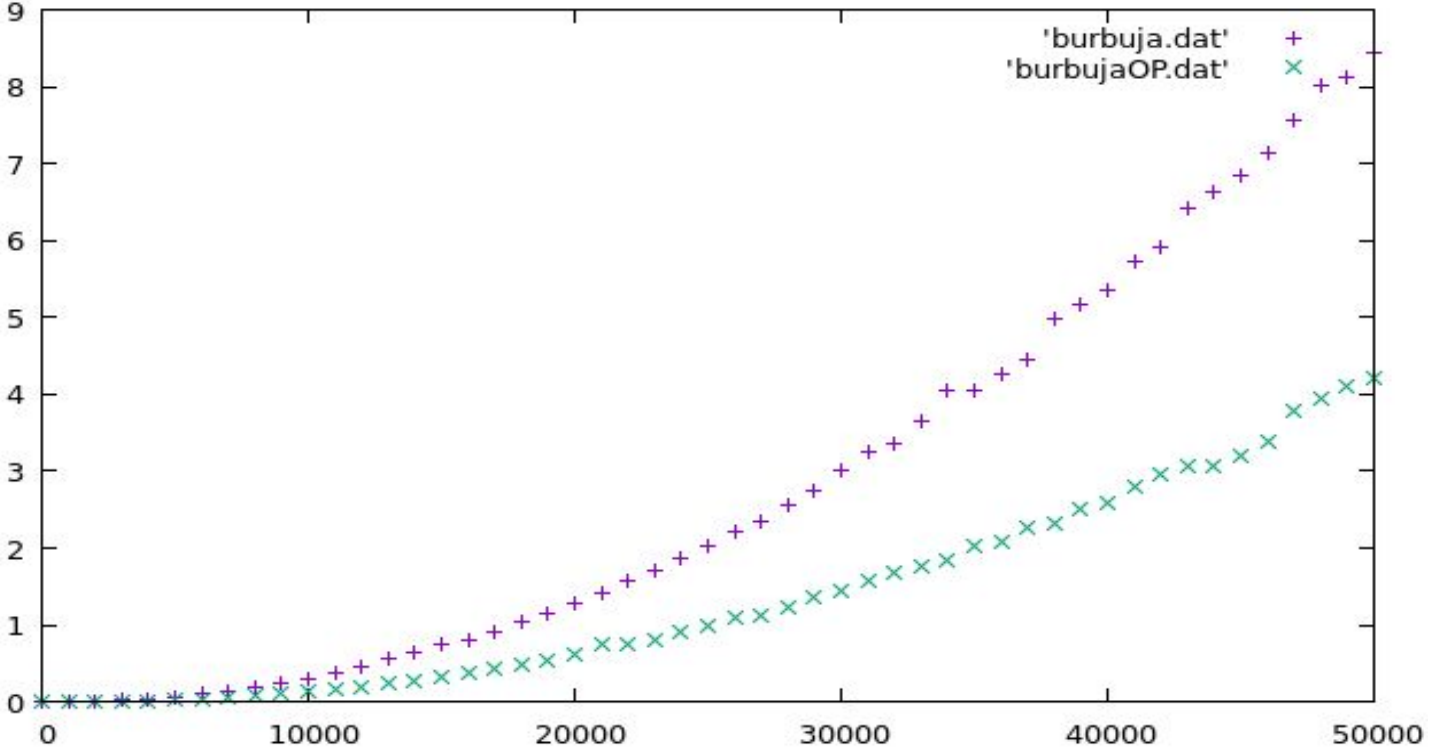
Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



5. Optimización

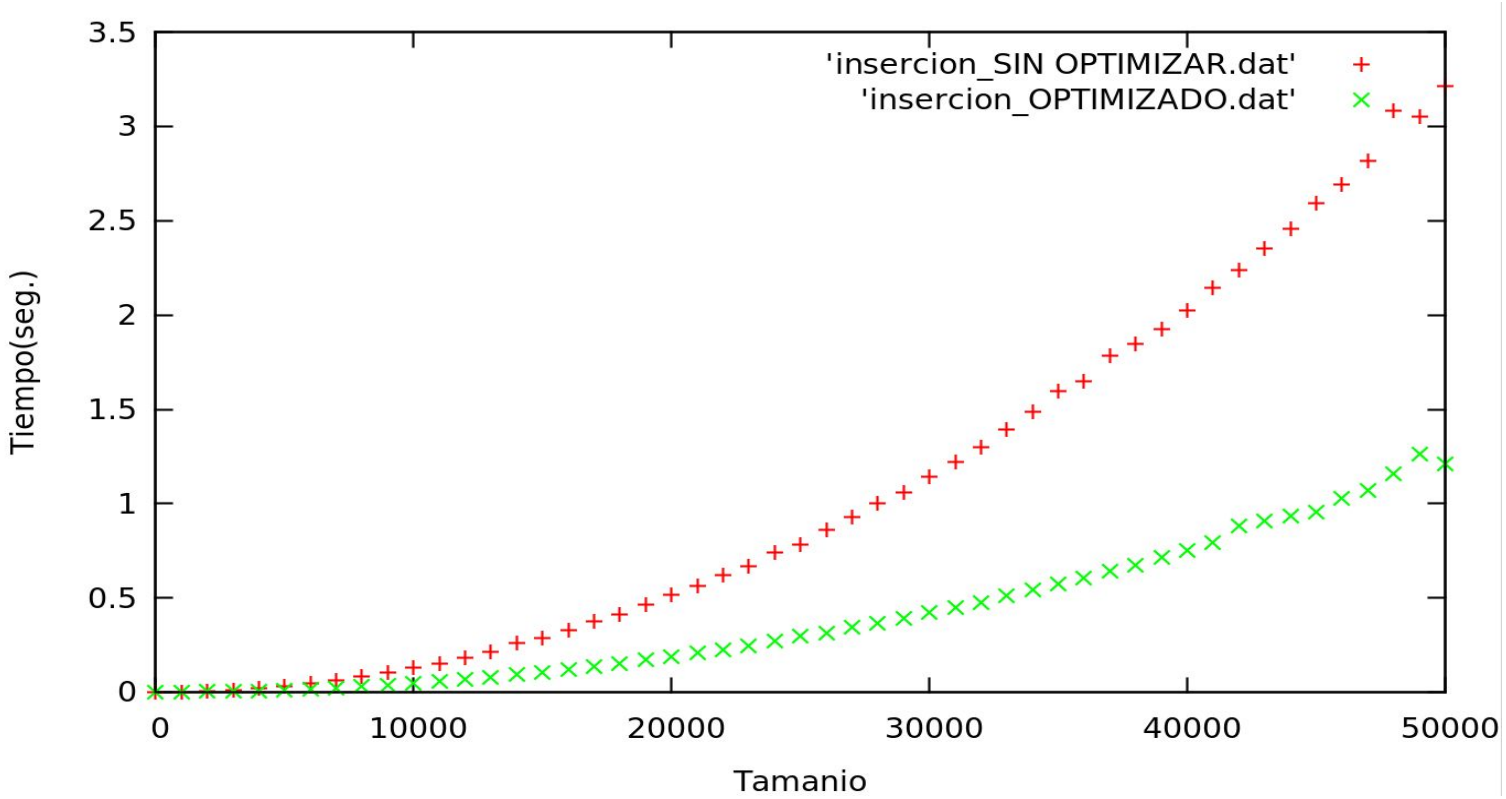
5.2.1.1 Burbuja

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



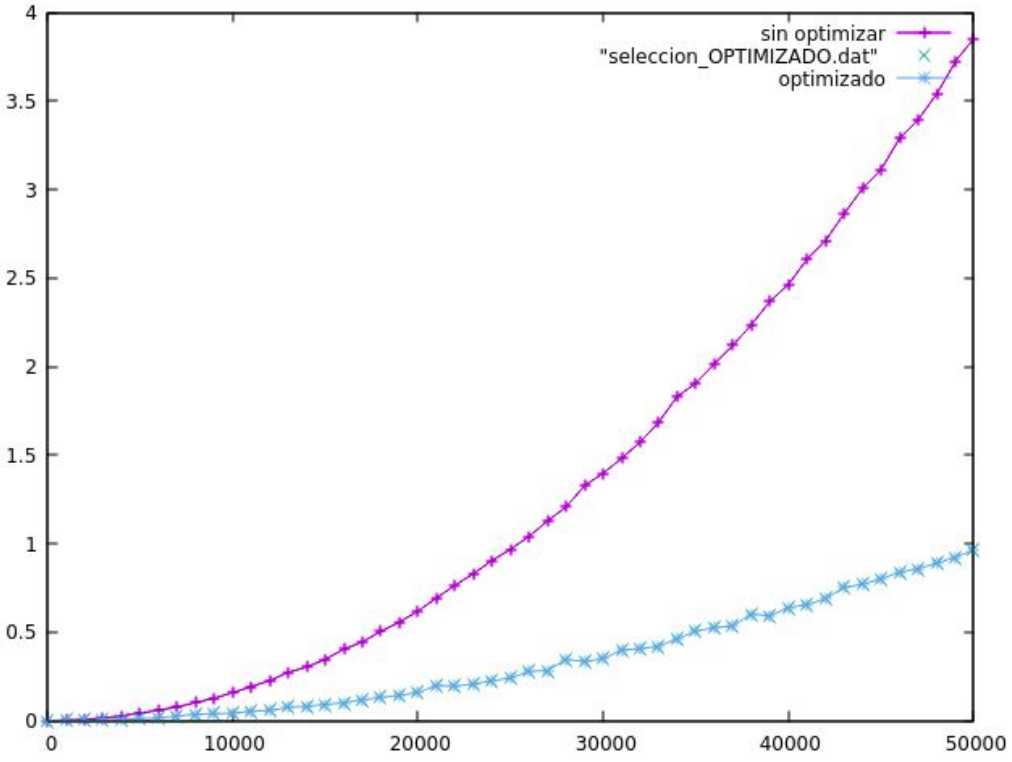
5.2.1.2 Inserción

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



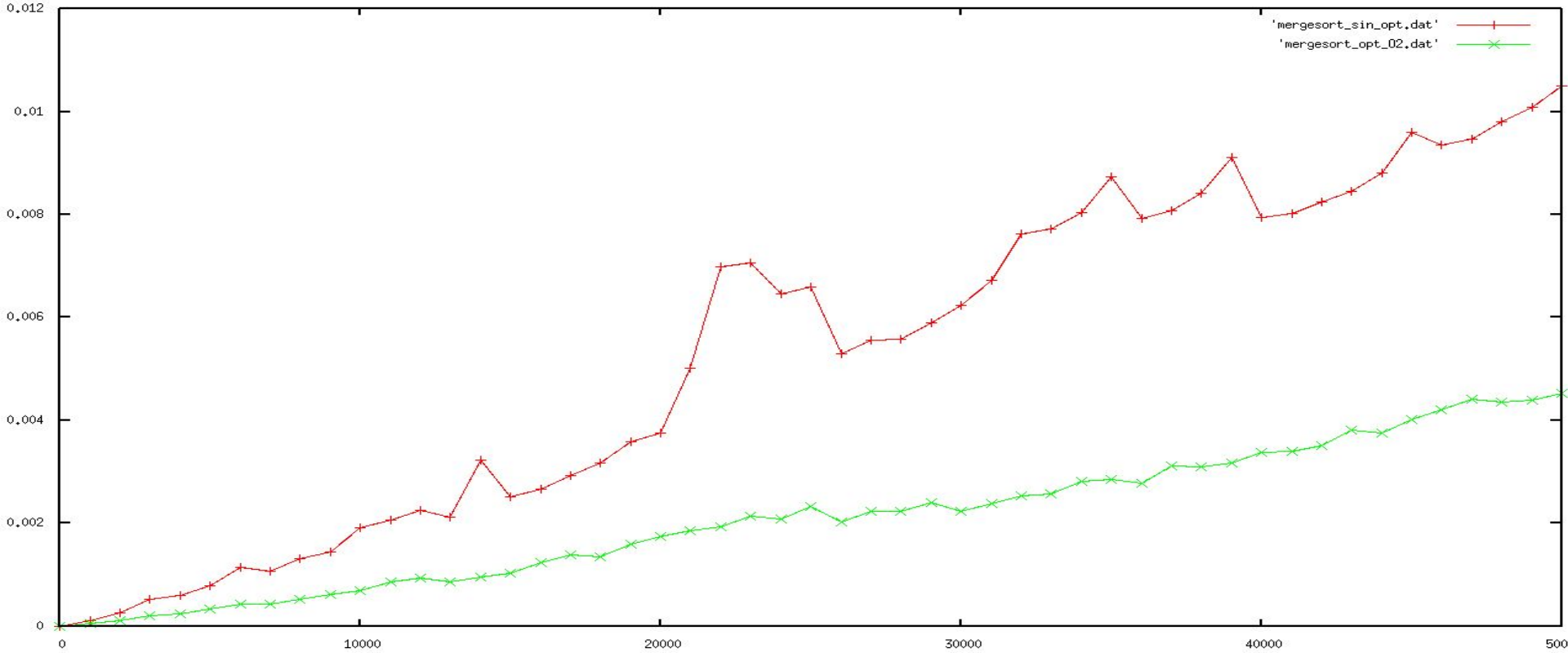
5.2.1.3 Selección

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



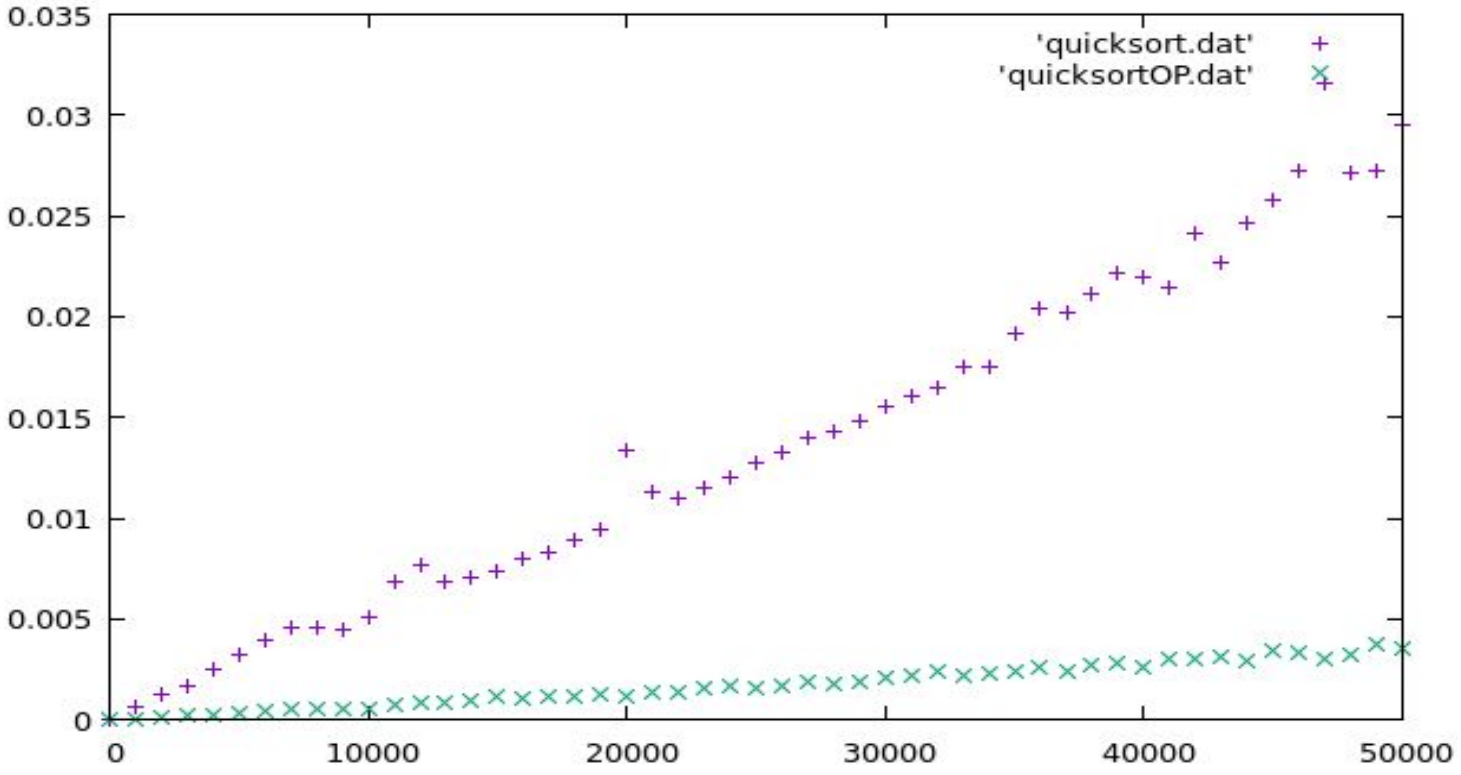
5.2.2.1 Mergesort

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz



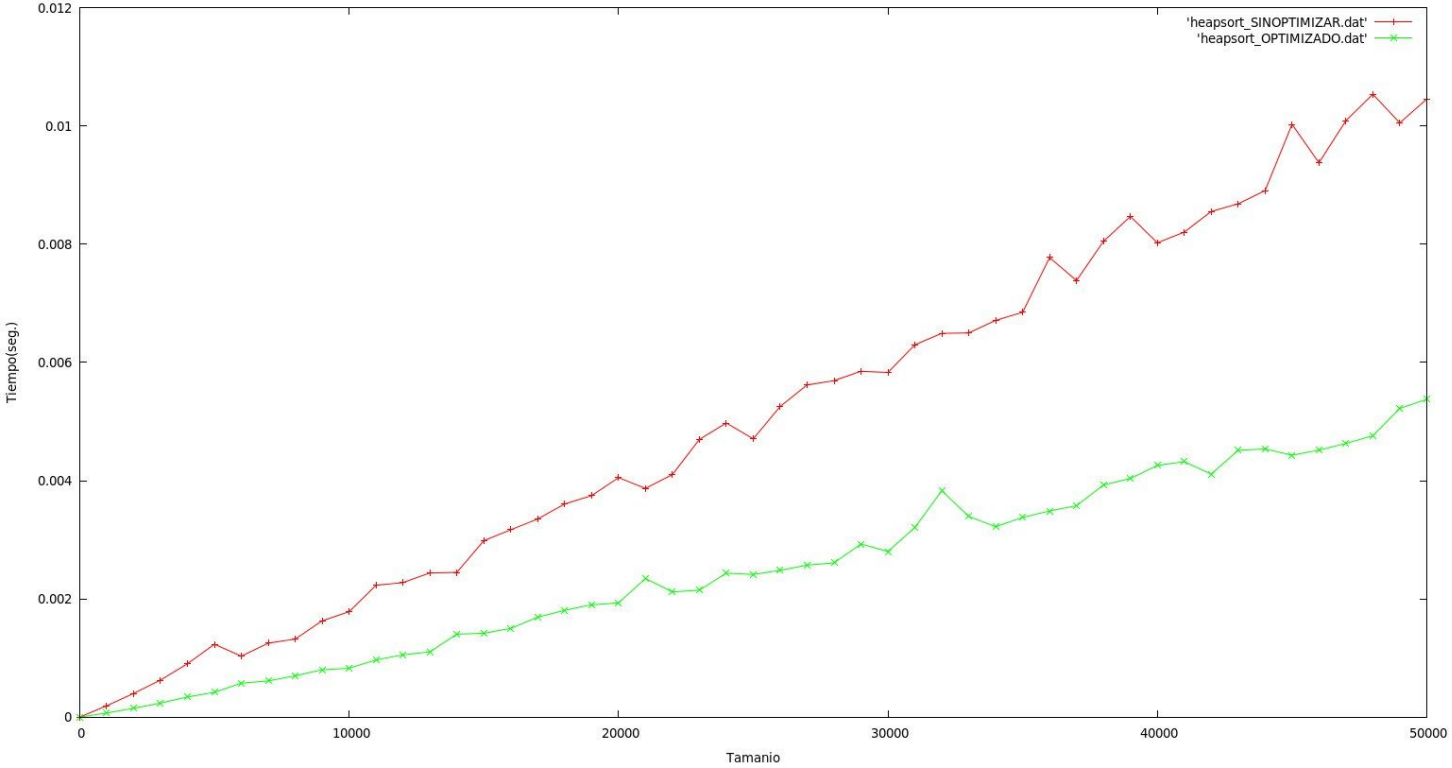
5.2.2.2 Quicksort

Procesadores	Intel Core i7-4500U CPU @ 1.80GHz



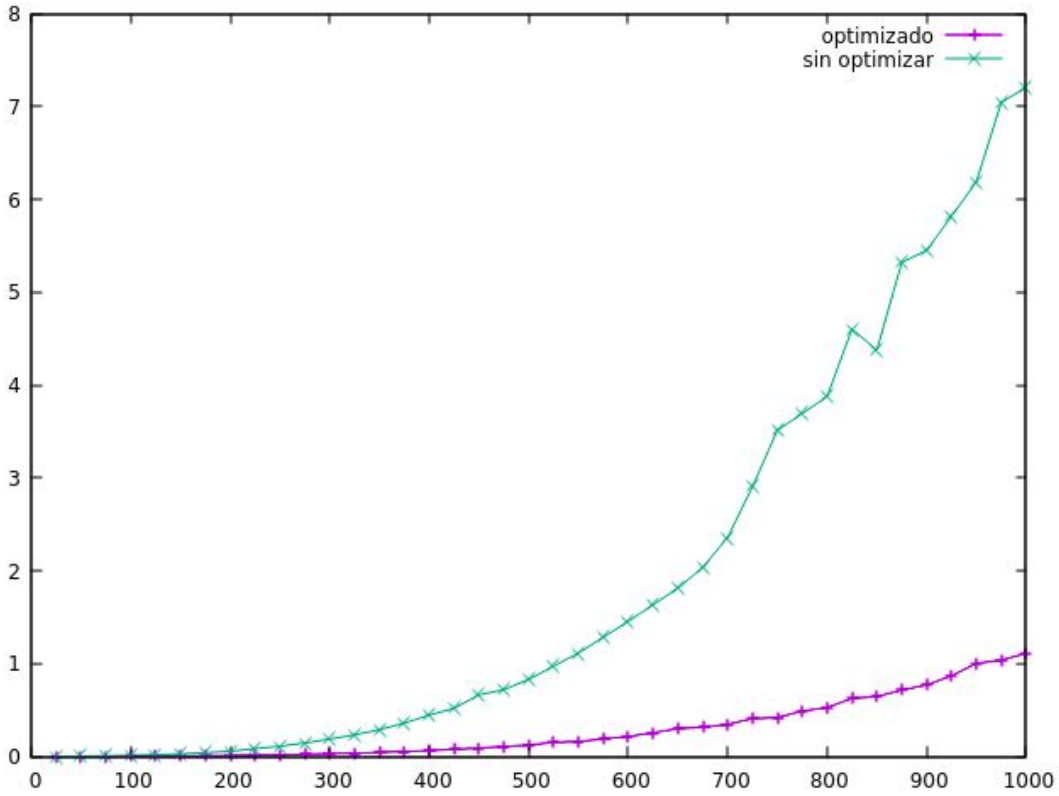
5.2.2.3 Heapsort

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



5.3 Algoritmo de Floyd

Procesadores	Intel Core i5-5200U CPU @ 2.20GHz



5.4 Hanoi

Procesadores	Intel Core i7-4700MQ CPU @ 2.40GHz