

OBSERVACIONES DE LA PRÁCTICA

Estudiante 1 (Nathalia Quiroga) Cod 202013212

Estudiante 2 (David Valderrama) Cod 201910987

	Máquina 1	Máquina 2
Procesadores	Core i5 8th Gen 1,8 GHz de dos núcleos	AMD Ryzen 7 4800H with Radeon Graphics, 2900 MHz, 8 Core(s) 16 Logical Processor(s)
Memoria RAM (GB)	8 GB 1600 MHz DDR3	8GB 3200 MHz
Sistema Operativo	MacOS (64-bit)	Windows (64-bit)

Tabla 1. Especificaciones de las máquinas para ejecutar las pruebas de rendimiento.

Maquina 1

Resultados

Carga de Catálogo PROBING

Factor de Carga (PROBING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
0.30	1324034,332	34248,097
0.50	1324034,332	34391,972
0.80	1324034,996	35171,034

Tabla 2. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando PROBING en la Maquina 1.

Carga de Catálogo CHAINING

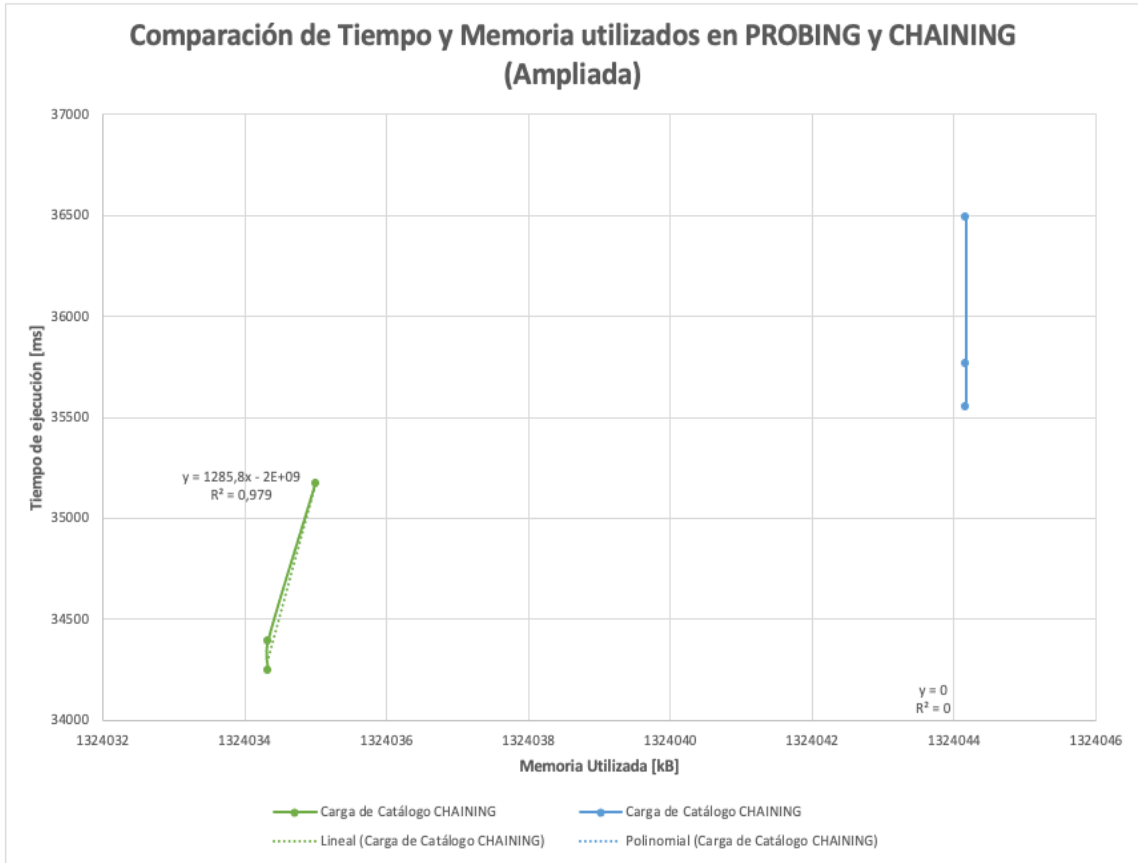
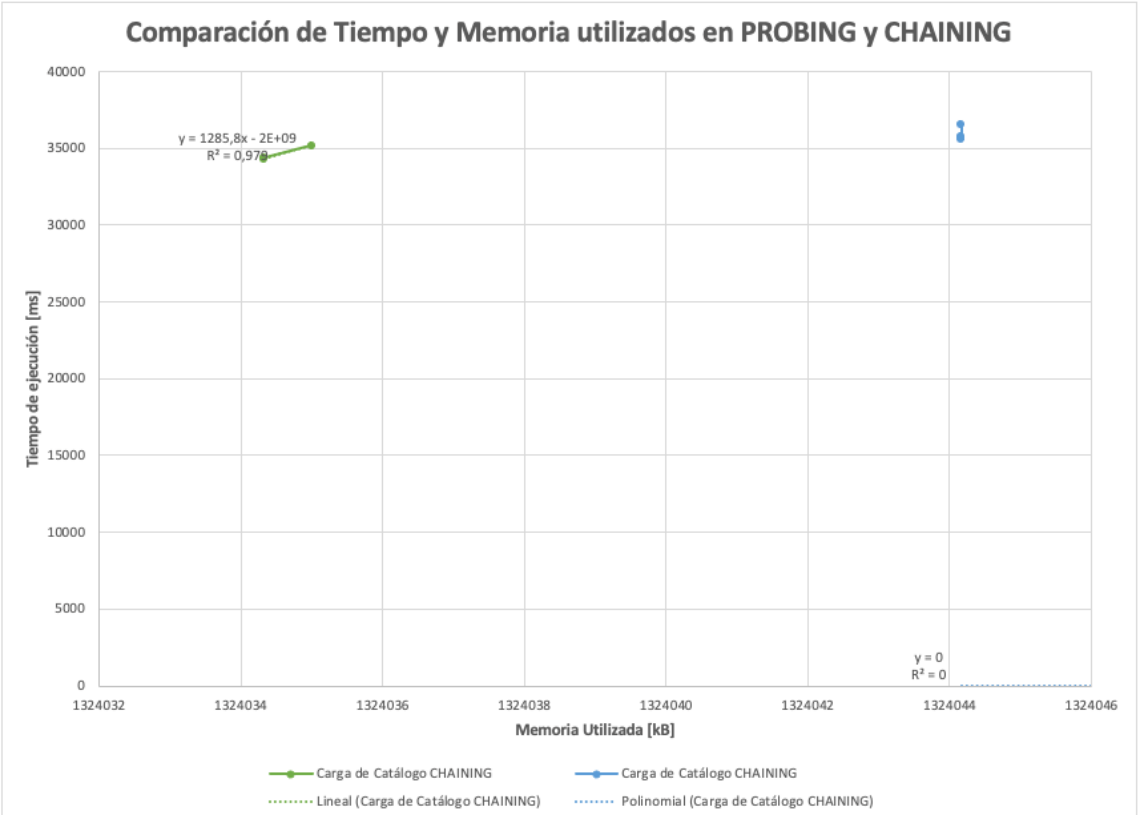
Factor de Carga (CHAINING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
2.00	1324044,164	35550,884
4.00	1324044,164	35766,408
6.00	1324044,164	36492,151

Tabla 3. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando CHAINING en la Maquina 1.

Graficas

La gráfica generada por los resultados de las pruebas de rendimiento en la **Maquina 1**.

- Comparación de memoria y tiempo de ejecución para PROBING y CHAINING



Maquina 2

Resultados

Carga de Catálogo PROBING

Factor de Carga (PROBING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
0.30	1324018.31	20130.373
0.50	1324017.983	19824.217
0.80	1324018.037	19553.6635

Tabla 4. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando PROBING en la Maquina 2.

Carga de Catálogo CHAINING

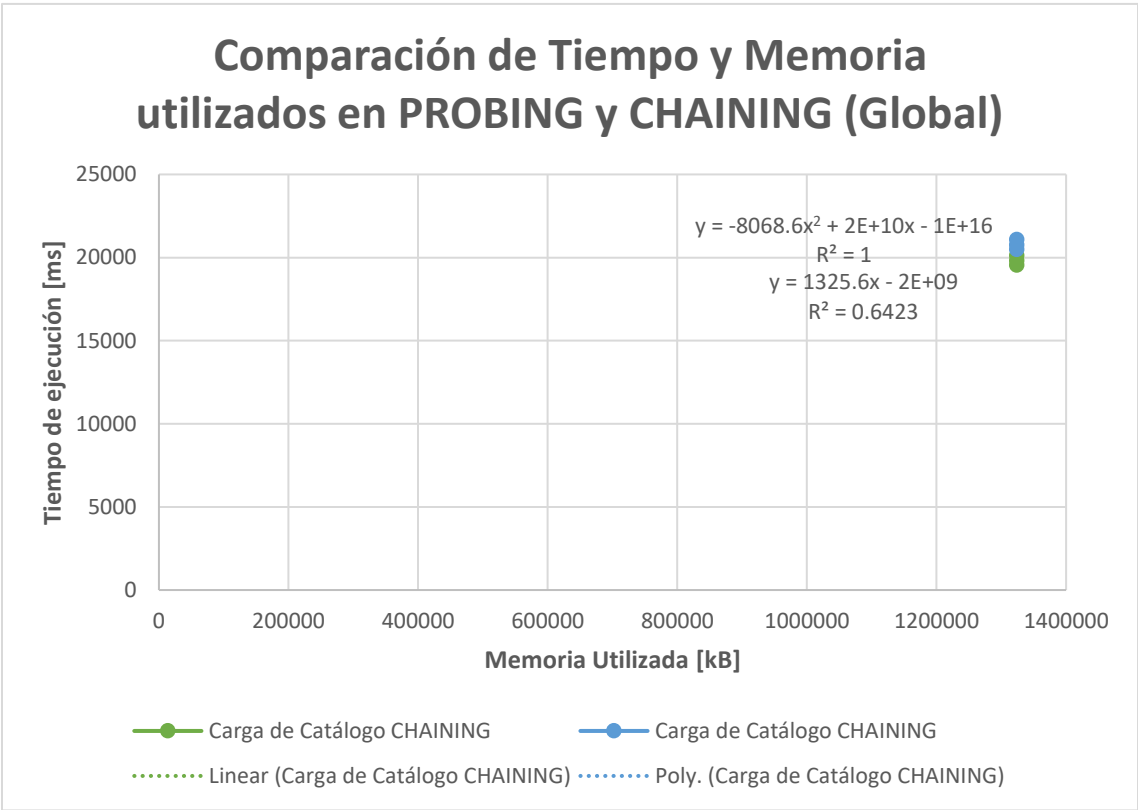
Factor de Carga (CHAINING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
2.00	1324024.922	20751.555
4.00	1324024.856	20490.361
6.00	1324025.195	21094.1005

Tabla 5. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando CHAINING en la Maquina 2.

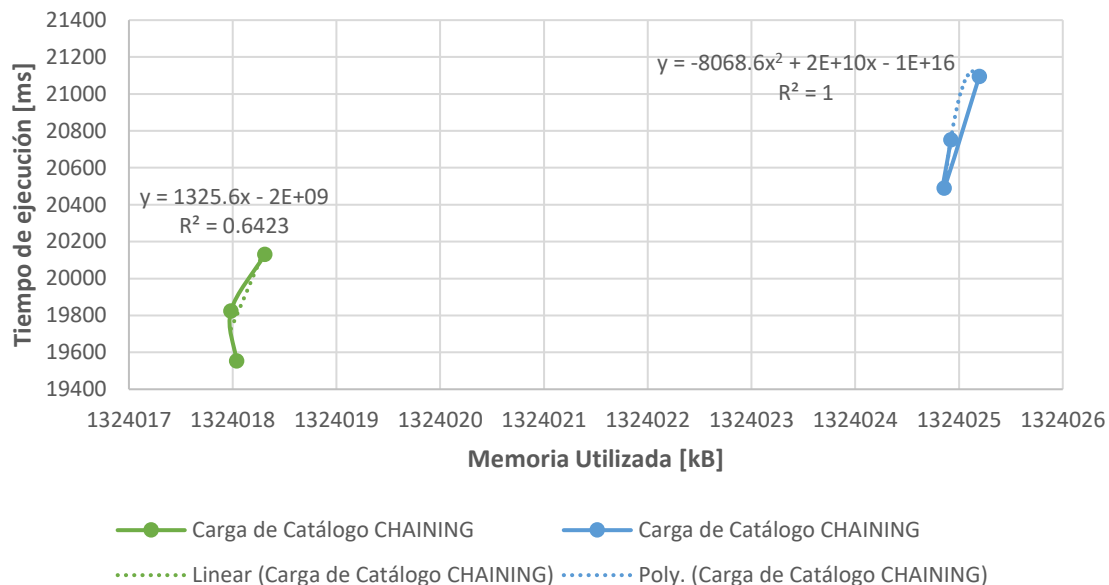
Graficas

La gráfica generada por los resultados de las pruebas de rendimiento en la **Maquina 2**.

- Comparación de memoria y tiempo de ejecución para PROBING y CHAINING



Comparación de Tiempo y Memoria utilizados en PROBING y CHAINING (Ampliada)



Preguntas de análisis

- 1) ¿Por qué en la función **getTime()** se utiliza **time.perf_counter()** en vez de la previamente conocida **time.process_time()**?
La función `perf_counter` mide el tiempo real que tarda un proceso, llamado también tiempo absoluto, similar a un cronómetro. Por otro lado, la función `process_time` es un valor derivado de la suma de tiempo de la CPU del sistema y del usuario del proceso actual, tomando en cuenta que la CPU no dedica el 100% del tiempo a ningún proceso dado.
- 2) ¿Por qué son importantes las funciones **start()** y **stop()** de la librería **tracemalloc**?
Porque nos permite hacer un rastreo de la asignación de memoria en Python. Así pues, `start()` inicia el rastreo sobre las posiciones de memoria y `stop()` guarda el registro y limpia todo lo que hizo `start()` sobre la memoria.
- 3) ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el factor de carga máximo para cargar el catálogo de videos?
Notamos un tiempo constante entre los factores de carga, esto se puede deber a la muestra de datos tan pequeña, ya que las colisiones presentadas son pocas y no difiere con el factor de carga. Naturalmente, los valores de tiempo cambian en magnitud dependiendo de la máquina, pero esto se puede atribuir a las especificaciones técnicas de las computadoras.
- 4) ¿Qué cambios percibe en el **consumo de memoria** al modificar el factor de carga máximo para cargar el catálogo de videos?

Existe un consumo de memoria constante entre los factores de carga, esto se puede deber a la muestra de datos tan pequeña. Naturalmente, los valores de consumo de memoria cambian en magnitud dependiendo de la máquina, pero esto se puede atribuir a las especificaciones técnicas de las computadoras.

- 5) ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el esquema de colisiones?, si los percibe, describa las diferencias y argumente su respuesta.

No percibimos alguna diferencia significativa debido a la poca cantidad de datos tomada para las pruebas, a pesar de esto, chaining ocupa un poco más de tiempo debido a la naturaleza de esta estructura de datos, ya que crea buckets haciendo un consumo mayor en tiempo.

- 6) ¿Qué cambios percibe en el **consumo de memoria** al modificar el esquema de colisiones?, si los percibe, describa las diferencias y argumente su respuesta.

No percibimos alguna diferencia significativa debido a la poca cantidad de datos tomada para las pruebas.