**OBSERVACIONES DEL LA PRACTICA**

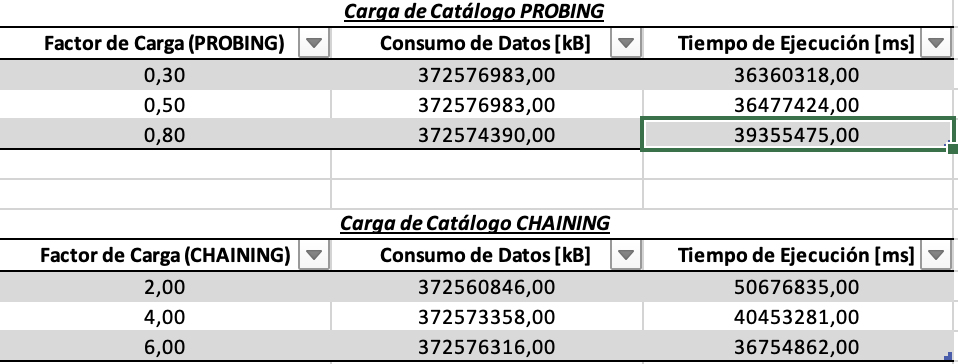
Juan Jose Sierra Cod 202013642

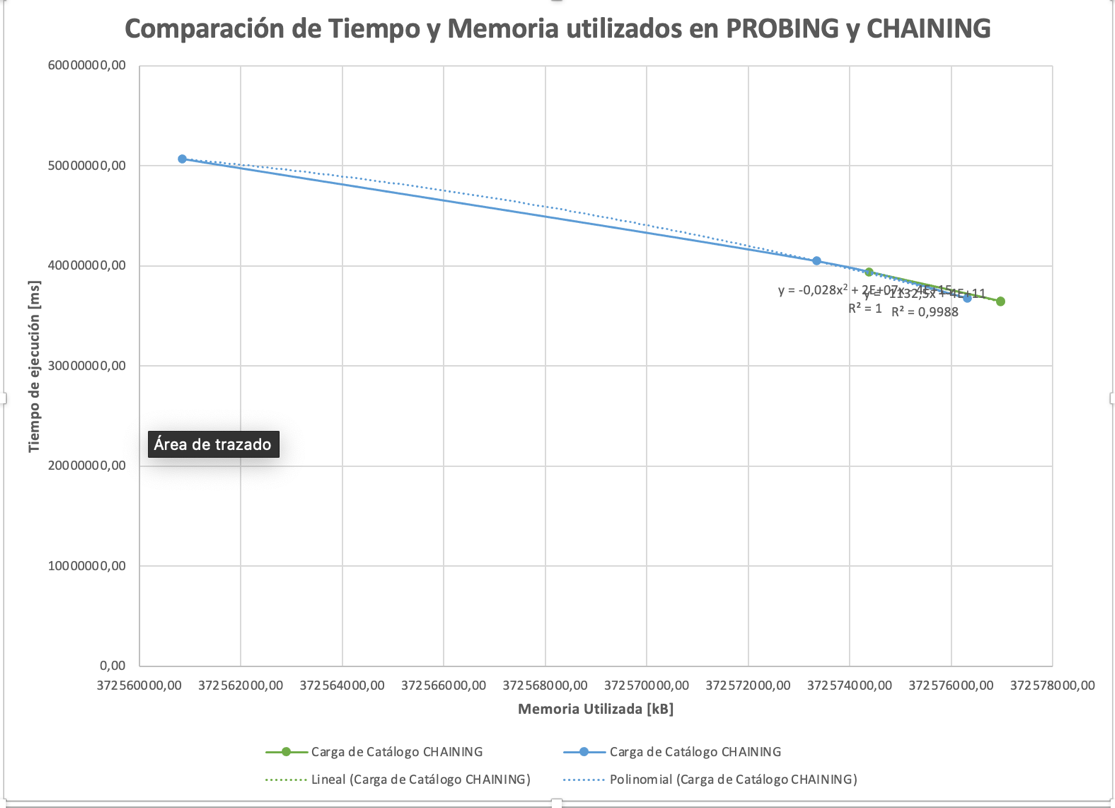
Juan Esteban Currea Cod 201922133

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz 1.60 GHz | 1,4 GHz Intel Core i5 de cuatro núcleos |
| Memoria RAM (GB) | 4.00 GB | 8 GB 2133 MHz LPDDR3 |
| Sistema Operativo | Windows 10 Pro | macOS Big Sur |

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

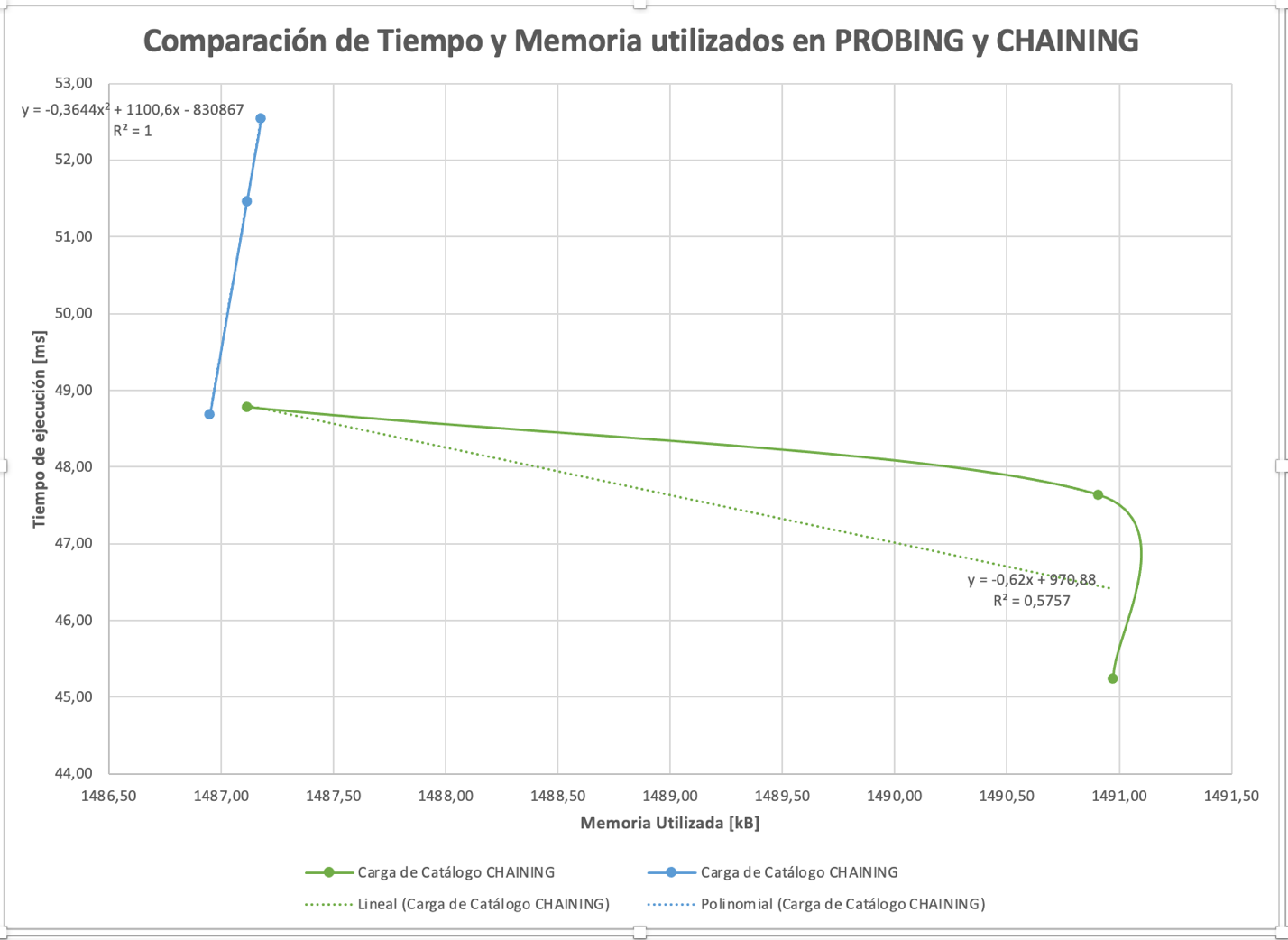
# **Maquina 1**





# **Maquina 2**

# 



# **Preguntas de análisis**

1. ¿Por qué en la función **getTime()** se utiliza **time.perf\_counter()** en ves de la previamente conocida **time.process\_time()**?
   1. Para este ejercicio se utiliza **time.perf\_counter() en ves de time.process\_time()** debido a que .pref\_counter nos permite calcular el tiempo de ejecución con máyor exactitud que .process\_time, pues utilizando .pref\_counter también se tiene el tiempo de ejecución “during sleep”. (Información basada en https://www.geeksforgeeks.org/time-perf\_counter-function-in-python/)
2. ¿Por qué son importantes las funciones **start()** y **stop()** de la librería **tracemalloc**?
   1. Tracemallo.start y tracemalloc.stop son importantes debido a que .start ayuda a iniciar la medición de la memoria utilizada desde un valor ya asignado en el momento en el que se corra el programa y, por el otro lado, .stop nos ayuda a encontrar el valor de la memoria utilizada al terminar de correr el programa. Entonces, ambas funciones son muy importantes ya que son la que nos ayudan a encontrar cuanta memoria se utiliza en total al correr el programa.
3. ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el factor de carga máximo para cargar el catálogo de videos?
   1. Utilizando PROBING podemos ver que el tiempo de ejecución aumenta en ambas maquinas mientras que, utilizando CHAINING solo en una de las dos máquinas vemos como el tiempo de ejecución aumenta, pues en la máquina 1 el tiempo disminuye.
4. ¿Qué cambios percibe en el **consumo de memoria** al modificar el factor de carga máximo para cargar el catálogo de videos?
   1. Utilizando PROBING podemos ver que el consumo de memoria disminuye en ambas maquinas mientras que, utilizando CHAINING podemos ver como en ambas maquinas aumenta el consumo de memoria.
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.
   1. En cuanto al tiempo de ejecución, podemos ver como, al utilizar PROBING este aumenta al ir aumentando su factor de carga máximo mientras que al utilizar CHAINING es completamente diferente en ambas máquinas, pues en la maquina 1 el tiempo disminuye mientras que en la maquina 2 este factor aumenta.
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.
   1. En cuento al consumo de memoria vemos un comportamniento similar utilizando ambos esquemas de colisiones, pues en las dos maquinas se evidencia un aumento de consumo de memoria al utilizar CHAINING mientras que vemos una disminución de este al utilizar PROBING.