**OBSERVACIONES DE LA PRACTICA**

Estudiante 1 Julián Castro del Valle-Cod 202020847

Estudiante 2 Tomás Otero Paris- Cod 202021733

|  |  |  |
| --- | --- | --- |
|  | Máquina 1 | Máquina 2 |
| Procesadores | Intel Core i5-3337U CPU @1.80GHz 1.80GHz | AMD Ryzen 5 6-core @3.6GHz |
| Memoria RAM (GB) | 8GB | 16GB |
| Sistema Operativo | Windows 10 Home | Windows 10 Home |

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 | 1070664.949 | 81417.329 |
| 0.50 | 962496.753 | 74690.220 |
| 0.80 | 848835.034 | 69056.831 |

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 | 916217.612 | 64573.957 |
| 4.00 | 897367.073 | 64338.814 |
| 6.00 | 894547.761 | 66394.304 |

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 | 904770.594 | 27672.289 |
| 0.50 | 800110.148 | 25714.509 |
| 0.80 | 690105.188 | 24138.222 |

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 | 755623.223 | 22031.816 |
| 4.00 | 737180.340 | 21852.318 |
| 6.00 | 734420.277 | 23446.780 |

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

# **Preguntas de análisis**

1. ¿Por qué en la función **getTime()** se utiliza **time.perf\_counter()** en ves de la previamente conocida **time.process\_time()**?

Se utiliza perf\_counter ya que es más exacto, al tomar en cuenta únicamente el tiempo de la operación particular que se quiere medir sin tener en cuenta el resto de operaciones del computador.

1. ¿Por qué son importantes las funciones **start()** y **stop()** de la librería **tracemalloc**?

Son importantes ya que determinan cuándo debe parar el conteo del tiempo transcurrido y la memoria utilizada.

1. ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el factor de carga máximo para cargar el catálogo de videos?

En probing, al aumentar el factor de carga máximo el tiempo de ejecución disminuye, mientras que en chaining éste tiende a aumentar.

1. ¿Qué cambios percibe en el **consumo de memoria** al modificar el factor de carga máximo para cargar el catálogo de videos?

En ambos casos, al aumentar el factor de carga máximo el consumo de memoria disminuye.

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

En ambas los tiempos son casi los mismos, no percibimos una diferencia significativa.

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

En probing el consumo de memoria es mayor que en chaining, lo cual contradice la teoría ya que se supone que probing debería ser más rápida al saber donde está cada posición de una lista.