**OBSERVACIONES DEL LA PRACTICA**

Miguel Perdomo - 201913791

Samuel Jaramillo - 202010768

Luisa Zambrano - 201914911

|  |  |  |  |
| --- | --- | --- | --- |
|  | Máquina 1 | Máquina 2 | Máquina 3 |
| Procesadores |  |  |  |
| Memoria RAM (GB) |  |  |  |
| Sistema Operativo |  |  |  |

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 Real @LP [ms]** |
| 0.1 |  |  |
| 0.5 |  |  |
| 0.7 |  |  |
| 0.9 |  |  |

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 Real @SC [ms]** |
| 2.00 |  |  |
| 4.00 |  |  |
| 6.00 |  |  |
| 8.00 |  |  |

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 Real @LP [ms]** |
| 0.1 |  |  |
| 0.5 |  |  |
| 0.7 |  |  |
| 0.9 |  |  |

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 Real @SC [ms]** |
| 2.00 |  |  |
| 4.00 |  |  |
| 6.00 |  |  |
| 8.00 |  |  |

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

# **Maquina 3**

## **Resultados**

|  |  |  |
| --- | --- | --- |
| ***Carga de Catálogo PROBING*** | | |
| **Factor de Carga (PROBING)** | **Consumo de Datos [kB]** | **Tiempo de Ejecución Real @LP [ms]** |
| 0.1 |  |  |
| 0.5 |  |  |
| 0.7 |  |  |
| 0.9 |  |  |

Tabla 6. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando PROBING en la Maquina 3.

|  |  |  |
| --- | --- | --- |
| ***Carga de Catálogo CHAINING*** | | |
| **Factor de Carga (CHAINING)** | **Consumo de Datos [kB]** | **Tiempo de Ejecución Real @SC [ms]** |
| 2.00 |  |  |
| 4.00 |  |  |
| 6.00 |  |  |
| 8.00 |  |  |

Tabla 7. Comparación de consumo de datos y tiempo de ejecución para carga de catálogo con el índice por categorías utilizando CHAINING en la Maquina 3.

## **Graficas**

La gráfica generada por los resultados de las pruebas de rendimiento en la **Maquina 3.**

* 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 vez de otras funciones como **time.process\_time()**?

Time.perf\_counter() da el tiempo absoluto que demora corriendo el código, mientras que la otra función devuelve el tiempo que demora el algoritmo pero ignorando el sleep time (que es tiempo de cambio entre los hilos del sistema operativo). Esto cambia el resultado dando un valor que no nos interese.

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

Es importante porque el start() define que comience a tomar el tiempo y el stop() lo detiene. Si no se detiene, la función se queda funcionando infitamente saturando el computador.

1. ¿Por qué no se puede medir paralelamente el **uso de memoria** y el **tiempo de ejecución** de las operaciones?

Medir la memoria implica llevar a cabo otras tareas en el fondo y esto afecta el tiempo de ejecución del algoritmo. Si medimos memoria el tiempo será mayor que el tiempo real de ejecuión (la medición de memoria afecta el resultado de tiempo). Por lo tanto, si solo quiero el tiempo de ejecución no debería tomar memoria al tiempo.

1. Teniendo en cuenta cada uno de los requerimientos del reto ¿Cuántos índices implementaría en el Reto? y ¿Por qué?
2. Según los índices propuestos ¿en qué caso usaría **Linear Probing** o **Separate Chaining** en estos índices? y ¿Por qué?
3. Dado el número de elementos de los archivos del reto (large), ¿Cuál sería el factor de carga para estos índices según su mecanismo de colisión?
4. ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el factor de carga máximo para cargar el catálogo de contenido Streaming?
5. ¿Qué cambios percibe en el **consumo de memoria** al modificar el factor de carga máximo para cargar el catálogo de contenido Streaming?
6. ¿Qué cambios percibe en el **tiempo de ejecución** al modificar el esquema de colisiones?, si los percibe, describa las diferencias y argumente su respuesta.
7. ¿Qué cambios percibe en el **consumo de memoria** al modificar el esquema de colisiones?, si los percibe, describa las diferencias y argumente su respuesta.
8. ¿Qué configuración de ideal ADT Map escogería para el **índice de años (“Año”)** ?, especifique el mecanismo de colisión, el factor de carga y el numero inicial de elementos.