j.montealegresQuniandes.edu.co
l.gomez1@uniandes.edu.co

Estudiante 1: Juan Montealegre Cod 202012723
Estudiante 2: Lina María Gómez Cod 201923531

LABORATORIO 7- COLISIONES

Preguntas Iniciales: LabCollision-S07

a. ¿Por qué en la función **`getTime()`** se utiliza **`time.perf_counter()`** en ves de la previamente conocida **`time.process_time()`**?

Se utiliza **time.perf_counter()** ya que este proporciona acceso al reloj con la mayor resolución disponible con el fin de hacer mediciones a corto plazo más precisas. En el caso específico de **time.process_time** este no tiene en cuenta el punto de referencia por lo que a diferencia de **time.perf_counter(), time.process_time()** no es tan rápido.

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

Tracemalloc es una librería que calcula la memoria utilizada entre dos instantes de tiempo.**Tracemalloc.start()** inicia el rastreo las asignaciones de la memoria en Python, gurdando un nframe del bloque más reciente. Por otro lado, **Tracemalloc.stop()** termina el rastreo de memoria y llama a la función take_snapshot() que captura instantáneo de los rastreos. Su importancia reside en que ayudan a saber la última cantidad de memoria que fue usada en la ejecución de una función.

Pruebas: RETO 2

Tabla 1. Parámetros Máquinas

	Máquina 1	**Máquina 2**
Procesadores	Intel® Core™ i7-8550U CPU @ 1.8GHZ, 2.00GHz	Intel® Core™ i5-8250U CPU @ 1.6GHZ, 1800 Mhz
Memoria RAM (GB)	16GB	12.0 GB
Sistema Operativo	Windows 10 Home-64-bits	Windows 10 Home-64-bits

1. Ejecute todas las pruebas de rendimiento propuestas en la misma máquina que reporto.

2. Cierre todas las aplicaciones que puedan afectar la toma de datos y sean **innecesarias** de su máquina, ej.: **Discord**, **OneDrive**, **Dropbox**, **Word** y en especial el navegador de internet (**Edge**, **Chrome**, **Firefox**, **Safari, ...**).

3. Se recomienda que cada prueba se ejecute por lo menos tres veces para poder obtener un promedio o consolidado consistente de las pruebas. Ej. Para un experimento para cargar el catálogo con factor de carga 0.6 se debe ejecutar por lo menos 3 veces (ideal 5 veces).

4. evitando errores dentro de la máquina como lo son actualizaciones o escaneos de seguridad, para registrar resultados adecuados en las tablas provistas.

5. Cada uno de los tiempos debe estas registrado en **milisegundos (ms)** y con 3 cifras decimales redondeado hacia arriba desde la mitad. Ej.: **43494.498587 ms** se aproxima a **43494.499 ms.**

PASO 6: Medir tiempo y memoria en el Reto No. 2

En esta sección evaluaremos el efecto en el uso de la memoria y tiempo de ejecución para **TAD Map** en la implementación del reto.

Para ello, integre a su código del Reto No. 2 los cambios estudiados en la primera parte de la práctica. Inclúyalos para medir el tiempo de ejecución y la memoria utilizada en la carga del catálogo de videos, recuerde que en la práctica anterior creo un índice por categorías con **TAD Map.**

Para ello siga las siguientes indicaciones:

1. Incluir las importaciones para medir tiempo y memoria en el **`controller.py.`**

2. Integrar las funciones **`getTime()`**, **`getMemory()`** y **`deltaMemory()`** en el **`controller.py`**

3. Modificar la su función de carga del catálogo de Videos para recuperar el tiempo de ejecución y la memoria utilizada en el proceso.

j.montealegresQuniandes.edu.co
l.gomez1@uniandes.edu.co

Estudiante 1: Juan Montealegre Cod 202012723
Estudiante 2: Lina María Gómez Cod 201923531

Al completar las modificaciones de su Reto No. 2, diligencie la Tabla 2 y la Tabla 3 variando el factor de carga y el mecanismo de colisiones entre **"PROBING"** y **"CHAINING"** para la Tabla de Hash en el índice de categorías (e.j.: *maptype='PROBING'* y *maptype='CHAINING'*).

MÁQUINA 1

Carga de Catálogo PROBING

Factor de Carga (PROBING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
0.30	1057143.299	30073.430
0.50	1057143.299	29087.256
0.80	1057143.299	28979.160

Tabla 2.Mediciones de tiempo y datos para diferentes factores de carga en PROBING.

Carga de Catálogo CHAINING

Factor de Carga (CHAINING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
2.00	1057152.523	30465.674
4.00	1057152.523	29860.350
6.00	1057152.523	28860.350

Tabla 3. Mediciones de tiempo y datos para diferentes factores de carga en CHAINING.

MÁQUINA 2

Carga de Catálogo PROBING

Factor de Carga (PROBING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
0.30	1307641.226	30839.818
0.50	1307641.226	29307.198
0.80	1307633.422	28291.447

Tabla 2.Mediciones de tiempo y datos para diferentes factores de carga en PROBING.

Carga de Catálogo CHAINING

Factor de Carga (CHAINING)	Consumo de Datos [kB]	Tiempo de Ejecución [ms]
2.00	1307636.769	29739.169
4.00	1307633.922	29748.519
6.00	1307633.922	29694.785

Tabla 3. Mediciones de tiempo y datos para diferentes factores de carga en CHAINING.

Preguntas Adicionales

j.montealegresQuniandes.edu.co
l.gomez1@uniandes.edu.co

Estudiante 1: Juan Montealegre Cod 202012723
Estudiante 2: Lina María Gómez Cod 201923531

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

Podemos percibir como al modificar el factor de carga, se ve afectado el tiempo de ejecución de este, entre menor sea el factor de carga mas se va a demorar en cargar la base de datos, esto puede ser debido a que entre más pequeño sea el factor de carga mayor va a ser m, el cual es el tamaño de la lista.

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

No se evidencia una diferencia en el consumo de la memoria al momento de cambiar el factor de carga maximo.

3. ¿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 lo que respecta al tiempo de ejecución a medida que se fue implementando la misma función en ambas máquinas se observó una reducción pequeña en el tiempo a medida que aumenta el factor de carga. No obstante, no hay una gran diferencia en cuanto el tiempo de ejecución al modificar el esquema de colisiones.

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

Para ambos casos, tanto linear probing como separate chaining no se observaron diferencias significativas entre el consumo de Datos [Kb]. Para la máquina 1 se observó en PROBING 1057143.299 kb y en CHAINING 1057152.523 kb (Chaining > Probing). Para el caso de la máquina 2, se observó un consumo de 1307641.226 kb en PROBING y 1307633.922 kb en CHAINING (Chaining < Probing). Por lo tanto, no se puede llegar a conclusiones definitivas cuando se modifica el esquema de colisiones.