Proyecto 1: Organización de Archivos

1- Logro del estudiante

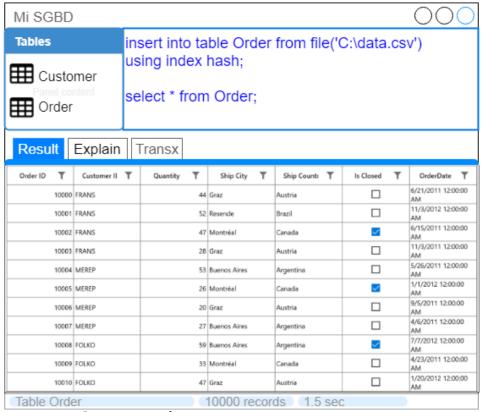
Entender y aplicar los algoritmos de almacenamiento de archivos físicos y lacceso concurrentel.

2- Enunciado

En grupos máximo de cinco integrantes, elegir un dominio de datos conformado por al menos dos archivos planos con datos reales (https://www.kaggle.com/datasets). Luego sobre ellos implementar tres técnicas de organización de archivos con sus principales operaciones: inserción, eliminación y búsqueda.

3- Requerimientos generales

- a. Implementar tres de las siguientes técnicas de organización de archivos en memoria secundaria.
 - 1. Sequential File / AVL File
 - 2. ISAM-Sparse Index / B+ Tree
 - 3. Extendible Hashing
- b. Operaciones que se deben implementar:
 - La búsqueda específica puede retornar más de un elemento que coincide con la key vector<Registro> search(T key)
 - 2. La búsqueda por rango retorna todos los registros que se encuentran entre las dos llaves de búsqueda vector<Registro> rangeSearch(T begin-key, T end-key)
 - Agregar un registro al archivo respetando la técnica de organización bool add(Registro registro)
 - 4. Proponer un algoritmo de eliminación para cada técnica bool remove(T key)
- c. Implementar un parser de código SQL a sentencias ejecutables. Eiemplos de sentencias:
 - create table Customer from file "C:\data.csv" using index hash("DNI")
 - Definir las estructuras del registro previamente en el codigo.
 - select * from Customer where DNI = x
 - select * from Customer where DNI between x and y
 - insert into Customer values (...)
 - delete from Customer where DNI = x
- d. Mostrar los resultados de forma amigable a usuario. Ver la siguiente GUI de referencia.



4- Conside racione s de la

implementación

- a. En el Sequential Index La función add(Registro registro) debe utilizar un espacio auxiliar para guardar los nuevos registros. Cuando el espacio auxiliar llegue a K registros, aplique un algoritmo de reconstrucción del archivo de datos manteniendo el orden físico de acuerdo a la llave seleccionada. Asegúrese de mantener los punteros actualizados.
- b. **En el ISAM** partir construyendo el índice estático hasta de 3 niveles a partir de un conjunto de datos. Para nuevas inserciones se generan los overflow pages (encadenamiento de páginas).
- c. **En el ISAM** de debe definir el factor de bloque tanto en las páginas de datos como en las páginas del índice.
- d. La función **rangeSearch** debe usar el índice para buscar el beginkey y luego recorrer los registros de acuerdo la estructura del índice. Esta función no es soportada por las técnicas de hashing.
- e. Use adecuadamente los conceptos de programación orientado a objetos y programación genérica para que el programa soporte cualquier dominio de datos.
- f. Para la interfaz gráfica pueden usar QT, Visual Studio C++, wxWidgets u otra librería gráfica.

5- Informe del provecto

- Archivo en Markdown / Wiki.
- Ortografía y consistencia en los párrafos.
- Al final del informe poner el video explicando el funcionamiento del programa, casos de uso, aspectos importantes de la implementación. El video no debe exceder los 10 minutos y deben participar todos los integrantes del grupo.

- Aspectos de evaluación e informe:

Item	Descripción
- Introducción (2 pts)	 Objetivo del proyecto. Descripción del dominio de datos a usar. Resultados que se esperan obtener al aplicar las técnicas de indexación.
- Técnicas Utilizadas. (9 pts)	 Describa brevemente las técnicas de indexación de archivos que ha elegido. Explique el algoritmo de inserción, eliminación y búsqueda (use gráficos para un mayor entendimiento) Se debe realizar un análisis comparativo teórico de las técnicas implementadas en base a los accesos a memoria secundaria tanto para las operaciones de inserción, búsqueda y eliminación. En el código debe estar optimizado en el manejo de memoria secundaria. Explicar claramente como se realizó el parser del SQL.
- Resultados Experimental es (5 pts)	 Cuadro y/o gráfico comparativo de desempeño de las técnicas de indexación de archivos sobre el dominio de datos. Tanto para inserción como para búsqueda. Considerar dos métricas: total de accesos a disco duro (read & write) y tiempo de ejecución en milisegundos. Discusión y análisis de los resultados experimentales.
- Pruebas de uso y presentación (4 pts)	 Presentar las pruebas de uso de la aplicación en interfaz gráfica amigable e intuitiva. Recuerde mostrar la funcionalidad del aplicativo en el video.

6- Entregable

- Los alumnos formarán grupos de máximo cinco integrantes.
- El código fuente del proyecto será alojado en GitHub Classroom.
- En el Canvas subir solo el **enlace público** del proyecto.

- La fecha límite de entrega es el 13 de Mayo de 2024.