|  |
| --- |
|  |
| **Recuperatorio: Informe** |
| **Ing. Informática**  **Algoritmos y Estructuras de Datos** |
| Bruno Olivera - CI: 4.867.466-7 |
| **Fecha de informe 09/07/2024** |
|  |

# Propuesta

En la Universidad Católica se desea desarrollar un sistema de bibliotecas para los alumnos, que les permitirá solicitar y devolver libros de manera eficiente, el sistema utilizara un árbol AVL para manejar el gran volumen de alumnos que solicitarán un libro, guardando en el cada alumno que tenga un libro prestado, dichos alumnos serán guardados por su número de cedula. AVL es una buena estructura para implementar en este caso ya que proporciona operaciones eficientes de búsqueda, inserción y eliminación, que es exactamente para lo que la usaremos.

Luego tendremos una estructura de Trie junto con un diccionario, para la búsqueda y gestión de libros, en la estructura de trie se almacenarán nombres de los libros, para tener una eficiente indización de estos, el usuario tendrá la capacidad de colocar en un buscador el nombre del libro, categoría o autor y el sistema le sugerirá los posibles resultados encontrados, este nombre se usará en un hashmap como clave, para poder obtener el valor, que será el libro.

Ya que la biblioteca debe cumplir con estrictas reglas de préstamos dictadas por la universidad cada alumno solo podrá llevarse consigo un solo libro. Un trie en este caso se utilizaría más que nada para la búsqueda rápida de nombres de libro, y un hashmap aseguraría un acceso rápido a los libros una vez encontrado el nombre.

Cada libro tendrá una ID, Nombre, Categoría, Autor y su disponibilidad, y cada Alumno contara con una CI, su nombre y el libro que actualmente tiene prestado.

La consigna consiste en crear un SistemaBiblioteca el cual contenga funcionalidades como:

**public void registrarAlumno(String cedula, String nombre)**

**public void registrarLibro(String nombreLibro, String categoria)**

**public boolean prestarLibro(String cedula, String nombreLibro)**

**public boolean devolverLibro(String cedula)**

**public Lista<Libro> buscarLibro(String libro)**

# Algoritmos Utilizados

## **Árbol BB**

Un árbol de búsqueda binaria es una estructura de datos en la que cada nodo tiene como máximo dos hijos. Los nodos se organizan de tal manera que, para cualquier nodo, todos los valores en su subárbol izquierdo son menores y todos los valores en su subárbol derecho son mayores.

## **Árbol AVL**

Un árbol AVL es un tipo de árbol de búsqueda binaria que se mantiene balanceado. Después de cada operación de inserción o eliminación, el árbol se reequilibra automáticamente para asegurar que la altura del árbol sea, garantizando operaciones eficientes.

## **Trie**

Un Trie es una estructura de datos en forma de árbol utilizada para almacenar un conjunto de cadenas. Cada nodo en un Trie representa un carácter de una cadena. Los Tries son eficientes para búsquedas de prefijos y sugerencias automáticas.

# Justificación

Se decidió usar un árbol AVL como estructura principal manejar a los alumnos que desean inscribirse en la librería para poder guardar los datos de los alumnos eficientemente, ya que estos datos podrían ser un gran volumen de datos.

Se decidió usar una estructura de trie para guardar los libros, en esta estructura se recorre el trie

# Conclusión

Dado el nuevo requerimiento de búsqueda por autor, categoría y nombre combinado, se decidió descartar la implementación de un hashmap para guardar el dato del libro, en vez de eso se decidió guardar el dato del libro en el nodo (es palabra).

En la función original el hashmap solo contenía el nombre del libro como clave y el trie devolvía dicho nombre para la búsqueda, con la nueva implementación esto ya no es necesario y tampoco eficiente, ya que hashmap solo puede guardar una sola clave, lo que se hacía difícil para guardar claves iguales como lo es una categoría de un libro, ya que de estas habría siempre más de una.