

**BIENVENIDO A BIBLIOTECA MAGICA  
ESTOS SON LOS TDA'S**

**MADELAYNE ANA MARIA PEREZ PEREZ  
202130171**

**LAB. DE ESTRUCTURA DE DATOS  
AUX. DAVID BARRERA**



## Introducción

El sistema “Biblioteca Mágica Alrededor del Mundo” gestiona una red interconectada de bibliotecas encantadas que comparten, buscan y trasladan libros a través de portales mágicos. Para lograr un funcionamiento eficiente, se implementaron estructuras de datos avanzadas (TAD's) desde cero, aplicando conceptos de organización, búsqueda y optimización algorítmica.

### ◆ Lista Enlazada y Lista Doble

#### Definición:

Estructura lineal donde los elementos (nodos) se enlazan mediante punteros a su siguiente o anterior elemento.

Dominio: Colecciones de libros.

#### Operaciones:

insertar(libro)

eliminar(isbn)

buscar(titulo)

mostrar()

#### Uso en el sistema:

Permite almacenar los libros por colección dentro de cada biblioteca, facilitando recorridos secuenciales y eliminaciones sin costo de reindexación.

#### Complejidad:

Inserción:  $O(1)$

Búsqueda secuencial:  $O(n)$

Eliminación:  $O(1)$  (si se tiene referencia)

### ◆ Pila (Stack)

#### Definición:

Estructura LIFO (Last In, First Out).

Dominio: Control de devoluciones y operaciones de deshacer.

#### Operaciones:

push(libro)

pop()

peek()

isEmpty()

#### Uso:

Se utiliza para almacenar libros devueltos y revertir registros erróneos mediante un sistema de rollback mágico.

Complejidad:

Todas las operaciones:  $O(1)$

#### ◆ Cola (Queue)

Definición:

Estructura FIFO (First In, First Out).

Dominio: Sistema de despacho de libros entre bibliotecas.

Operaciones:

enqueue(libro)

dequeue()

front()

isEmpty()

Uso:

Cada biblioteca tiene tres colas:

Cola de ingreso

Cola de traspaso

Cola de salida

Estas colas simulan el flujo de libros en tránsito según intervalos de procesamiento configurables.

Complejidad:

Encolado y desencolado:  $O(1)$

#### ◆ Árbol AVL

Definición:

Árbol binario de búsqueda balanceado que mantiene la altura equilibrada en cada inserción o eliminación.

Dominio: Búsqueda y ordenamiento por título.

Operaciones:

insertar(título, libro)

eliminar(título)

buscar(título)

rotarIzquierda()

rotarDerecha()

Uso:

Optimiza las búsquedas por título, manteniendo el catálogo siempre ordenado alfabéticamente.

Complejidad:

Inserción:  $O(\log n)$

Eliminación:  $O(\log n)$

Búsqueda:  $O(\log n)$

### ◆ Árbol B

Definición:

Árbol balanceado multi-nivel usado para almacenar registros de libros por año.

Dominio: Filtrado por año de publicación.

Operaciones:

insertar(anio, libro)

buscarRango(inicio, fin)

eliminar(anio)

Uso:

Permite consultas eficientes sobre libros en rangos de años o generaciones mágicas.

Complejidad:

Inserción:  $O(\log n)$

Búsqueda:  $O(\log n)$

Eliminación:  $O(\log n)$

### ◆ Árbol B+

Definición:

Extensión del Árbol B con todos los datos almacenados en los nodos hoja enlazados secuencialmente.

Dominio: Clasificación por género.

Operaciones:

insertar(genero, libro)

buscar(genero)

listarGenero(genero)

Uso:

Facilita la agrupación de libros por género y permite recorridos ordenados entre categorías.

Complejidad:

Inserción:  $O(\log n)$

Búsqueda:  $O(\log n)$

Recorrido:  $O(n)$

### ◆ Tabla Hash

Definición:

Estructura dispersa con función hash que convierte el ISBN mágico en índice de almacenamiento.

Dominio: ISBN único de cada libro.

Operaciones:

insertar(isbn, libro)

buscar(isbn)

eliminar(isbn)

Uso:

Permite búsquedas directas por ISBN con manejo de colisiones mediante encadenamiento.

Complejidad:

Inserción:  $O(1)$  promedio

Búsqueda:  $O(1)$  promedio

Eliminación:  $O(1)$  promedio

### Grafo Ponderado

Definición:

Conjunto de vértices (bibliotecas) y aristas (conexiones) con pesos que representan tiempo o costo de traslado.

Dominio: Red de bibliotecas.

Operaciones:

agregarNodo(biblioteca)

agregarConexion(origen, destino, peso)

caminoMinimo(origen, destino)

Uso:

Determina rutas mágicas óptimas entre bibliotecas utilizando algoritmos de Dijkstra o Floyd.

Complejidad:

Dijkstra:  $O(V^2)$  o  $O(E \log V)$  con heap

Inserción nodo/arista:  $O(1)$