

Aplicación de estructuras de datos en bibliotecología: Uso de árboles para la organización y recuperación de información

Alumnos:

Melisa Martellini(melisainesmartellini@gmail.com)

Fabrizio Puccio (pucciofabrizio1@gmail.com)

Materia:

Programación I

Profesor:

Sebastián Bruselario

Tutora:

Virginia Cimino

Fecha de Entrega: 09 de junio 2025

1. Introducción

Se analizará la relación entre la bibliotecología y las estructuras de datos, con el objetivo de entender cómo la organización, almacenamiento y recuperación de información dependen de modelos y sistemas eficientes de gestión. Considerando su impacto en la accesibilidad, automatización y optimización de la información.

2. Marco teórico

Bibliotecología: Definición y Objetivos

La bibliotecología, también conocida como ciencia de la información, es la disciplina que estudia las bibliotecas y los servicios de información en general. Se enfoca en la organización, el acceso y la difusión del conocimiento, empleando diversas herramientas y metodologías para el registro, búsqueda y recuperación de información.

El uso de herramientas y tecnologías de la información y la comunicación (TIC) ha transformado la gestión bibliotecaria, permitiendo a los profesionales administrar la información de manera más eficiente y efectiva.

- **Sistemas manuales y automatizados:** Facilitan el registro, búsqueda y recuperación de datos, mejorando el acceso a la información.
- **Automatización de bibliotecas:** Optimiza la gestión documental, agiliza procesos y amplía la accesibilidad para los usuarios.

La bibliotecología se relaciona estrechamente con otras áreas del conocimiento, como la informática, la comunicación y la educación, enriqueciendo sus métodos y enfoques para la gestión del conocimiento

Según la American Library Association (ALA), las bibliotecas pueden clasificarse en las siguientes categorías:

- **Escolares:** Destinadas a apoyar el aprendizaje en instituciones educativas.
- **Públicas:** Accesibles a toda la comunidad, con recursos diversos.
- **Nacionales:** Representan la memoria documental de un país.
- **Especializadas:** Orientadas a áreas específicas del conocimiento.

En bibliotecas y archivos, la información se administra mediante diversos procesos técnicos que garantizan su organización y accesibilidad:

- **Catalogación:** Permite la creación de registros bibliográficos detallados, facilitando la identificación de materiales, empleando reglas de estándar internacional para la organización de registros bibliográficos.
- **Clasificación:** Organiza los recursos según criterios específicos, mejorando la estructura del acervo documental.
- **Recuperación de información:** Optimiza la búsqueda y el acceso a los datos, asegurando que los usuarios puedan localizar los documentos con rapidez.

Para gestionar eficazmente estos procesos, las bibliotecas utilizan software especializado que les permite administrar su acervo documental de manera eficiente y automatizada.

Árboles en la gestión de información

Los árboles son estructuras jerárquicas que facilitan la organización de datos en sistemas bibliotecarios y de archivos. Algunos tipos incluyen:

- **Árboles binarios:** Organizan información de manera estructurada, permitiendo búsquedas rápidas.

- Árboles B y B+: Utilizados en bases de datos para gestionar grandes volúmenes de información de manera eficiente.
- Trie: Aplicado en motores de búsqueda y sistemas de autocompletado, mejorando la recuperación de datos.

Los árboles binarios cumplen un rol fundamental en la organización, almacenamiento y recuperación de información dentro de sistemas bibliotecarios. Dado a su estructura jerárquica, permiten optimizar procesos de búsqueda y clasificación, haciendo que el acceso a los datos sea más rápido y eficiente.

En un catálogo automatizado, por ejemplo, se puede utilizar un árbol binario de búsqueda (Binary Search Tree, BST) para indexar los libros según su número de clasificación. Este tipo de estructura permite que, al realizar una consulta, el sistema recorra el árbol y localice el documento deseado en mucho menos tiempo que mediante una búsqueda secuencial.

Los árboles binarios destacan por su capacidad para localizar información de manera eficiente. Cada nodo puede tener hasta dos hijos, lo cual permite aplicar algoritmos de búsqueda que reducen significativamente el número de comparaciones necesarias para encontrar un elemento.

Más allá de los árboles binarios, los árboles como estructura de datos en general son ideales para representar sistemas jerárquicos. Se componen de un nodo raíz (root) del que se derivan los demás nodos, y cada uno puede tener uno o varios hijos. Una propiedad esencial es que no contienen ciclos, condición clave para representar estructuras de clasificación ordenadas. Este tipo de organización resulta especialmente útil en bibliotecas, donde los esquemas de clasificación como la Clasificación Decimal Dewey (CDD) o la Clasificación Decimal Universal (CDU) pueden modelarse como árboles jerárquicos, permitiendo representar con claridad las relaciones entre temas y subtemas.

Propiedades clave de los árboles binarios:

- Raíz: nodo inicial.
- Hojas: nodos sin hijos.
- Altura: número de niveles desde la raíz hasta la hoja más profunda.
- Subárboles: cualquier nodo junto a sus descendientes forma un subárbol.

Caso Práctico: Aplicación de un Árbol Binario en la gestión bibliográfica

Las bibliotecas digitales requieren sistemas eficientes para la organización y recuperación de sus registros bibliográficos. Para lograrlo, se implementa un árbol de búsqueda binaria, donde cada nodo representa un libro, clasificado por su título.

Cada nodo contiene información relevante sobre un libro, incluyendo su título, autor, año de publicación y editorial. La organización de los nodos se basa en el orden alfabético del título, de manera que:

- Los libros con títulos menores se ubican en el subárbol izquierdo.
- Los libros con títulos mayores se ubican en el subárbol derecho.

Esta estructura permite una búsqueda eficiente y rápida, evitando recorridos innecesarios dentro del catálogo.

Metodología

Para representar los libros dentro del árbol binario, se utiliza la clase `Nodo`, que define los atributos esenciales de cada elemento. Se han desarrollado funciones específicas para:

- Crear el árbol binario.
- Insertar nuevos nodos (libros).
- Realizar recorridos para visualizar el contenido de manera ordenada.

El método de recorrido utilizado es inorden, lo que garantiza que los libros se presenten en orden ascendente de acuerdo con sus títulos.

Para comparar los títulos de los libros, Python permite ordenar cadenas de texto siguiendo el orden ASCII, mediante operadores como `<` y `>`. Esto facilita la inserción de cada nodo en la posición adecuada dentro del árbol.

Desarrollo / Implementación

A continuación, se presenta el código completo

Este código define la clase `Nodo` con 4 atributos, que representa cada libro dentro de un árbol binario de búsqueda.

```
class Nodo:
    def __init__(self, titulo, autor, anio, editorial):
        self.titulo = titulo
        self.autor = autor
        self.anio = anio
        self.editorial = editorial
        self.izquierda = None
        self.derecha = None
```

```
class BibliotecaC: #define la estructura del árbol.
    def __init__(self): #crea una nueva instancia del árbol
        self.raiz = None #nodo inicial vacío
```

[illegible]

```
#Compara el título del nuevo libro con el nodo actual.
#Si es menor, lo coloca a la izquierda; si es mayor, a la derecha.
def _insertar_recursivo(self, nodo, nuevo_nodo):
    if nuevo_nodo.titulo < nodo.titulo:
        if nodo.izquierda is None:
            nodo.izquierda = nuevo_nodo
        else:
            self._insertar_recursivo(nodo.izquierda, nuevo_nodo) #recursión para seguir bajando en el árbol
    else:
        if nodo.derecha is None:
            nodo.derecha = nuevo_nodo
        else:
            self._insertar_recursivo(nodo.derecha, nuevo_nodo) #recursión para seguir bajando en el árbol
```

```
#Busca un libro por su título dentro del árbol.
def buscar(self, titulo):
    return self._buscar_recursivo(self.raiz, titulo)
```

[illegible]

```
def _inorden(self, nodo): #Recorre en orden -inorden-
    if nodo:
        self._inorden(nodo.izquierda)
        print(f"Título: {nodo.titulo}, Autor: {nodo.autor}, Año: {nodo.anio}, Editorial: {nodo.editorial}")
        self._inorden(nodo.derecha)
```

```
#Se insertan los datos de los libros.
```

```
biblioteca = BibliotecaC()
biblioteca.insertar("La metamorfosis", "Franz Kafka", 1996, "Losada")
biblioteca.insertar("Un elefante ocupa mucho espacio", "Elsa Bornemann", 2004, "Alfaguara,")
biblioteca.insertar("El eternoauta", "Héctor Germán Oesterheld", 2023, "Planeta")
biblioteca.insertar("Hamlet", "William Shakespeare", 1969, "Salvat")
```

```
# Buscar un libro ingresado por el usuario
titulo_buscar = input("Ingrese el título del libro").strip().lower() #Elimina espacios y convertir a minúsculas
resultado = biblioteca.buscar(titulo_buscar) # para realizar la búsqueda
if resultado and resultado.titulo.lower() == titulo_buscar:
    print(f"\nLibro encontrado: {resultado.titulo}, Autor: {resultado.autor}, Año: {resultado.anio}, Editorial: {resultado.editorial}")
else:
    print("\nLibro no encontrado")
```

Conclusión

Este código presenta una implementación eficiente de un árbol binario de búsqueda, diseñado para la administración de una biblioteca. Permite la inserción, búsqueda y visualización de libros de manera ordenada y optimizada, garantizando una gestión estructurada de los registros bibliográficos.

Bibliografía

Comunidad Baratz. (2020, 21 de mayo). *Las distintas clasificaciones y tipologías de bibliotecas según UNESCO, INE, IFLA y ALA*. Comunidad Baratz. <https://www.comunidadbaratz.com/blog/las-distintas-clasificaciones-y-tipologias-de-bibliotecas-segun-unesco-ine-ifla-y-ala/>

González Guitián, M. V., & Molina Piñeiro, M. (2008). La evolución de la ciencia y la tecnología: revisión de sus indicadores. ACIMED, 18(6). Recuperado de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352006000200007

Sabor, J. E., Cagnoli, R. V., Revello, L. H., & Soto, S. (1984). *Manual de bibliotecología* (2.^a ed., ampliada y actualizada). Editorial Kapelusz Mexicana

Perrone, G., Murillo Madrigal, A., & González del Valle, L. (2002). *Introducción bibliotecológica: procesos técnicos y soportes de información* (Cuadernillo 1). Biblioteca Nacional de Maestros. Recuperado de http://www.bnm.me.gov.ar/redes_federales/publicaciones/doc/cuadernillo_1.pdf

Salvador Oliván, J. A., & Arquero Áviles, R. (2006). Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación. *Investigación bibliotecológica*, 20(41), 13-43. Recuperado de https://www.scielo.org.mx/scielo.php?pid=S0187-358X2006000200002&script=sci_arttext

IBM. (s.f.). *Organización y contenido del árbol de archivos*. Recuperado el 31 de mayo de 2025, de <https://www.ibm.com/docs/es/aix/7.2.0?topic=concepts-organization-contents-file-tree>

MSMK University. (2024, 4 de septiembre). *¿Qué es el árbol binario?* MSMK University. <https://msmk.university/binary-tree/#:~:text=%C2%BFQu%C3%A9%20es%20el%20binary%20tree,estructuras%20jer%C3%A1rquicas%20en%20la%20inform%C3%A1tica>

Valverde, J. F. (s.f.). *Árboles binarios de búsqueda*. Recuperado el 31 de mayo de 2025, de https://ccia.ugr.es/~jfv/ed1/tedi/cdrom/docs/arb_BB.htm

Downey, A., Elkner, J., & Meyers, C. (2002). *Aprenda a pensar como un programador con Python* (1.^a ed.). Green Tea Press. Recuperado de <https://argentinaenpython.com/quiero-aprender-python/aprenda-a-pensar-como-un-programador-con-python.pdf>