

Ejercicio: Sistema de Gestión de Biblioteca

Objetivo

Desarrollar un programa en C que permita gestionar una biblioteca utilizando estructuras (structs) y arrays para almacenar la información de los libros.

Descripción

Se requiere implementar un sistema básico para administrar una biblioteca que permita:

- Crear nuevos libros
- Listar todos los libros disponibles
- Prestar un libro
- Modificar la información de un libro
- Buscar un libro por su ISBN utilizando búsqueda binaria

Estructura de datos

Deberás definir una estructura `Libro` con los siguientes campos:

- Título (cadena de caracteres)
- ISBN (cadena de caracteres)
- Autor (cadena de caracteres)
- Editorial (cadena de caracteres)
- Estado de préstamo (valor booleano o entero: 0 para disponible, 1 para prestado)

Requisitos técnicos

1. Utilizar un array fijo con capacidad para 100 libros.
2. Implementar un menú interactivo que se ejecute en un bucle `do-while` hasta que el usuario elija salir.
3. Cargar al inicio del programa 3 libros ya predefinidos (hardcodeados) para facilitar las pruebas.
4. Para buscar un libro por ISBN, implementar el algoritmo de búsqueda binaria (esto requiere mantener los libros ordenados por ISBN).
5. No es necesario implementar la función de borrar libros.

Funcionalidades a implementar

1. Crear un libro

- Solicitar al usuario los datos del nuevo libro (título, ISBN, autor, editorial).
- Agregar el libro al final del array.
- Validar que el ISBN no esté duplicado.

2. Listar los libros

- Mostrar en pantalla todos los libros almacenados con su información completa.
- Indicar si el libro está disponible o prestado.

3. Prestar un libro

- Solicitar el ISBN del libro a prestar.
- Ordenar el array y luego buscar el libro utilizando búsqueda binaria.
- Cambiar su estado a "prestado" si está disponible.
- Mostrar un mensaje adecuado si el libro ya está prestado o no existe.

4. Modificar un libro

- Solicitar el ISBN del libro a modificar.
- Ordenar el array y luego buscar el libro utilizando búsqueda binaria.
- Permitir modificar título, autor o editorial (el ISBN no debe modificarse).

5. Buscar un libro

- Ordenar el array por ISBN.
- Implementar búsqueda binaria por ISBN.
- Mostrar la información completa del libro si se encuentra.

Estructura del menú

===== SISTEMA DE BIBLIOTECA =====

1. Crear un libro
2. Listar todos los libros
3. Prestar un libro
4. Modificar un libro
5. Buscar un libro por ISBN
6. Salir

Seleccione una opción:

Consejos y pistas

- Para la búsqueda binaria, primero ordene el array por ISBN antes de realizar la búsqueda (pueden usar un algoritmo como bubble sort, selection o insertion...).
- Agregar nuevos libros al final del array para simplificar la implementación (podríamos ir agregando en el orden correcto, pero no!).
- Utilizá funciones separadas para cada una de las operaciones principales.
- Recuerda validar los datos ingresados por el usuario para evitar comportamientos inesperados.

Ejemplo de hardcodeo inicial

```
// Ejemplo de cómo hardcodear 3 libros iniciales
void inicializarBiblioteca(Libro biblioteca[], int *cantidadLibros) {
    strcpy(biblioteca[0].titulo, "Don Quijote de la Mancha");
    strcpy(biblioteca[0].isbn, "1234");
    strcpy(biblioteca[0].autor, "Miguel de Cervantes");
    strcpy(biblioteca[0].editorial, "Planeta");
    biblioteca[0].prestado = 0;

    strcpy(biblioteca[1].titulo, "Cien años de soledad");
    strcpy(biblioteca[1].isbn, "2345");
    strcpy(biblioteca[1].autor, "Gabriel García Márquez");
    strcpy(biblioteca[1].editorial, "Planeta");
    biblioteca[1].prestado = 0;

    strcpy(biblioteca[2].titulo, "El principito");
    strcpy(biblioteca[2].isbn, "3456");
    strcpy(biblioteca[2].autor, "Antoine de Saint-Exupéry");
    strcpy(biblioteca[2].editorial, "Salamandra");
    biblioteca[2].prestado = 0;

    *cantidadLibros = 3;
}
```

Estructura base del programa

```
#include <stdio.h>
#include <string.h>

typedef struct {
    char titulo[100];
    char isbn[20];
    char autor[50];
    char editorial[50];
    int prestado; // 0: disponible, 1: prestado
} Libro;

// Prototipos de funciones
void inicializarBiblioteca(Libro biblioteca[], int *cantidadLibros);
void mostrarMenu();
void crearLibro(Libro biblioteca[], int *cantidadLibros);
void listarLibros(Libro biblioteca[], int cantidadLibros);
```

```
void prestarLibro(Libro biblioteca[], int cantidadLibros);  
void modificarLibro(Libro biblioteca[], int cantidadLibros);  
void ordenarBibliotecaPorISBN(Libro biblioteca[], int cantidadLibros); // Función para  
ordenar antes de buscar  
int buscarLibroBinario(Libro biblioteca[], int cantidadLibros, char isbn[]);
```