Roll NO:27

NAME : Mali Anjali Prakash

DIV: FYMCA-A

Assignment no -2

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

// Define structure for Book

typedef struct Book {

    char title[100];

    char author[100];

    struct Book\* next;

} Book;

// Function to create a new book node

Book\* createBook(char title[], char author[]) {

    Book\* newBook = (Book\*)malloc(sizeof(Book));

    strcpy(newBook->title, title);

    strcpy(newBook->author, author);

    newBook->next = NULL;

    return newBook;

}

// Function to add a book to the list

void addBook(Book\*\* head, char title[], char author[]) {

    Book\* newBook = createBook(title, author);

    if (\*head == NULL) {

        \*head = newBook;

    } else {

        Book\* temp = \*head;

        while (temp->next != NULL) {

            temp = temp->next;

        }

        temp->next = newBook;

    }

    printf("Book added successfully.\n");

}

// Function to remove a book by title

void removeBook(Book\*\* head, char title[]) {

    if (\*head == NULL) {

        printf("The list is empty.\n");

        return;

    }

    Book\* temp = \*head;

    Book\* prev = NULL;

    // Check if the book is the first one

    if (strcmp(temp->title, title) == 0) {

        \*head = temp->next;

        free(temp);

        printf("Book removed successfully.\n");

        return;

    }

    // Search for the book to remove

    while (temp != NULL && strcmp(temp->title, title) != 0) {

        prev = temp;

        temp = temp->next;

    }

    // If the book is not found

    if (temp == NULL) {

        printf("Book not found.\n");

        return;

    }

    prev->next = temp->next;

    free(temp);

    printf("Book removed successfully.\n");

}

// Function to display all books

void displayBooks(Book\* head) {

    if (head == NULL) {

        printf("No books available.\n");

        return;

    }

    Book\* temp = head;

    while (temp != NULL) {

        printf("Title: %s\n", temp->title);

        printf("Author: %s\n\n", temp->author);

        temp = temp->next;

    }

}

// Function to search for a book by title

void searchBook(Book\* head, char title[]) {

    Book\* temp = head;

    while (temp != NULL) {

        if (strcmp(temp->title, title) == 0) {

            printf("Book found!\n");

            printf("Title: %s\n", temp->title);

            printf("Author: %s\n\n", temp->author);

            return;

        }

        temp = temp->next;

    }

    printf("Book not found.\n");

}

// Function to count total number of books

int countBooks(Book\* head) {

    int count = 0;

    Book\* temp = head;

    while (temp != NULL) {

        count++;

        temp = temp->next;

    }

    return count;

}

int main() {

    Book\* head = NULL; // Initialize the head of the linked list

    int choice;

    char title[100], author[100];

    do {

        printf("\nMenu:\n");

        printf("1. Add Book\n");

        printf("2. Remove Book\n");

        printf("3. Display All Books\n");

        printf("4. Search for a Book\n");

        printf("5. Count Total Books\n");

        printf("6. Exit\n");

        printf("Enter your choice: ");

        scanf("%d", &choice);

        getchar(); // To consume the newline character left by scanf

        switch (choice) {

            case 1:

                printf("Enter book title: ");

                fgets(title, sizeof(title), stdin);

                title[strcspn(title, "\n")] = 0;  // Remove the newline character

                printf("Enter book author: ");

                fgets(author, sizeof(author), stdin);

                author[strcspn(author, "\n")] = 0;

                addBook(&head, title, author);

                break;

            case 2:

                printf("Enter book title to remove: ");

                fgets(title, sizeof(title), stdin);

                title[strcspn(title, "\n")] = 0;

                removeBook(&head, title);

                break;

            case 3:

                displayBooks(head);

                break;

            case 4:

                printf("Enter book title to search: ");

                fgets(title, sizeof(title), stdin);

                title[strcspn(title, "\n")] = 0;

                searchBook(head, title);

                break;

            case 5:

                printf("Total number of books: %d\n", countBooks(head));

                break;

            case 6:

                printf("Exiting...\n");

                break;

            default:

                printf("Invalid choice. Please try again.\n");

        }

    } while (choice != 6);

    return 0;

}







