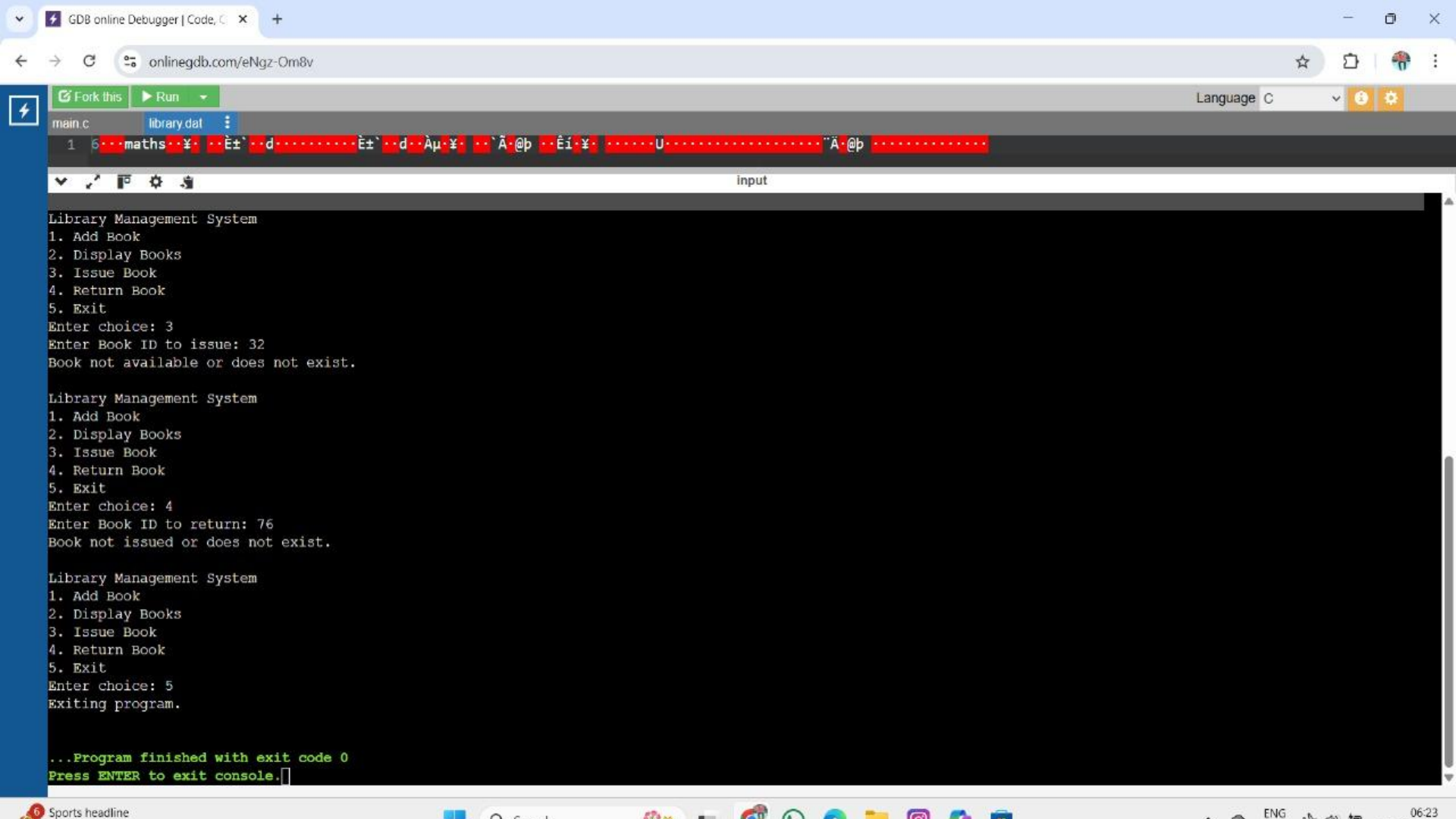
```
Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 1
Enter Book ID: 54
Enter Book Title: maths
Book added successfully.

Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 2

Available Books:
--------------------------
ID: 54
Title: maths
Status: Available


Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 3
Enter Book ID to issue: 32
Book not available or does not exist.
```


GDB online Debugger | Code,
onlinegdb.com/eNgz-Om8v
Fork this
Run
Language C
main.c
library.dat
6 maths ¥ È± d È± d Àµ ¥ `Ã @þ Êí ¥ U ¨Ã @þ
input
Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 3
Enter Book ID to issue: 32
Book not available or does not exist.
Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 4
Enter Book ID to return: 76
Book not issued or does not exist.
Library Management System
1. Add Book
2. Display Books
3. Issue Book
4. Return Book
5. Exit
Enter choice: 5
Exiting program.
...Program finished with exit code 0
Press ENTER to exit console.
Sports headline
ENG
06:23

```c
#include <stdlib.h>
#include <string.h>

struct Book {
    int id;
    char title[100];
    int available;  // 1 = available, 0 = issued
};

void addBook() {
    FILE *fp = fopen("library.dat", "ab");
    struct Book b;

    printf("Enter Book ID: ");
    scanf("%d", &b.id);
    printf("Enter Book Title: ");
    getchar(); // clear buffer
    fgets(b.title, 100, stdin);
    b.title[strcspn(b.title, "\n")] = 0; // remove newline
    b.available = 1;

    fwrite(&b, sizeof(b), 1, fp);
    fclose(fp);

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

void displayBooks() {
    FILE *fp = fopen("library.dat", "rb");
    struct Book b;

    printf("\nAvailable Books:\n");
    printf("-------------------------\n");
    while (fread(&b, sizeof(b), 1, fp)) {
        printf("ID: %d\nTitle: %s\nStatus: %s\n\n", b.id, b.title, b.available ? "Available" : "Issued");
    }
```

```c
    fclose(fp);
}

void issueBook() {
    FILE *fp = fopen("library.dat", "rb+");
    struct Book b;
    int id, found = 0;

    printf("Enter Book ID to issue: ");
    scanf("%d", &id);

    while (fread(&b, sizeof(b), 1, fp)) {
        if (b.id == id && b.available) {
            b.available = 0;
            fseek(fp, -sizeof(b), SEEK_CUR);
            fwrite(&b, sizeof(b), 1, fp);
            found = 1;
            printf("Book issued successfully.\n");
            break;
        }
    }

    if (!found) {
        printf("Book not available or does not exist.\n");
    }

    fclose(fp);
}

void returnBook() {
    FILE *fp = fopen("library.dat", "rb+");
    struct Book b;
    int id, found = 0;

    printf("Enter Book ID to return: ");
    scanf("%d", &id);
```

```c
            printf("Book returned successfully.\n");
            break;
        }
    }

    if (!found) {
        printf("Book not issued or does not exist.\n");
    }

    fclose(fp);
}

int main() {
    int choice;

    do {
        printf("\nLibrary Management System\n");
        printf("1. Add Book\n");
        printf("2. Display Books\n");
        printf("3. Issue Book\n");
        printf("4. Return Book\n");
        printf("5. Exit\n");
        printf("Enter choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1: addBook(); break;
            case 2: displayBooks(); break;
            case 3: issueBook(); break;
            case 4: returnBook(); break;
            case 5: printf("Exiting program.\n"); break;
            default: printf("Invalid choice.\n");
        }
    } while (choice != 5);

    return 0;
}
```