DATA STRUCTURES

Assignment - 1

Problem Statement : Library Book Borrowing System

Coding :-

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

// Structure for Book Borrow Record
typedef struct Borrow {
    int studentID;
    int bookID;
    char title[100];
    char date[20];
    struct Borrow *next;
} Borrow;

Borrow *head = NULL; // main list

// Function to create new node
Borrow * createBorrow (int studentID, int BookID, char
title[], char date[]) {
    Borrow newNode =
(Borrow) malloc (sizeof (Borrow));
```

```
    newNode -> StudentID = StudentID;
    newNode -> bookID = bookID;
    Strcpy (newNode -> title, title);
    Strcpy (newNode -> date, date);
    newNode -> next = NULL;
    return newNode;
}

// Issue a newbook (Insert node at end)
void issueBook (int StudentID, int bookID, char title[], char date[])
{
    Borrow * newNode = createBorrow (StudentID, bookID,
                                     title, date);
    if (!head)
    {
        head = newNode;
    }
    else
    {
        Borrow * temp = head;
        while (temp -> next) temp = temp -> next;
        temp -> next = newNode;
    }
    printf ("Book issued succesfully! \n");
}

// Return a book (Delete by Book ID)
void returnBook (int bookID) {
```

```
    Borrow * temp = head, * prev = NULL;
    while (temp) {
    if (temp → bookID == bookID)
    {
      if (prev) prev → next = temp → next;
      else head = temp → next;
      free (temp);
      printf ("Book with ID %d returned
successfully! \n", bookID);
      return;
    }
    prev = temp;
    temp = temp → next;
  }
  printf (" Book with ID %. not found. \n", bookID);
}
// Search borrowed books by Student ID
void SearchByStudent (int StudentID)
{
  Borrow * temp = head;
  int found = 0;
  while (temp)
  {
    if (temp → StudentID == StudentID)
    {
```

```
        printf ("Student %d borrowed Book %d | %s ) Date : %s
                                                    \n",
                temp → studentID, temp → bookID,
                temp → title, temp → date);
        found = 1;
    }
    temp = temp → next;
}
if (!found) printf (" No records for student ID
                %d.\n", studentID);
}
// Display all borrowing history
void displayAll ()
{
    if (!head)
    {
        printf ("No borrowing records.\n");
        return;
    }
    Borrow * temp = head;
    printf ("\n --- Borrowing Records --- \n");
    while (temp)
    {
        printf ("Student ID : %d | BookID : %d | Title : %s
                Date : %s\n",
    temp → studentID, temp → bookID, temp → title,
                temp → date);
    temp = temp → next;
}}
```

```
// clone list (for audits)
Borrow * cloneList (Borrow * src)
{
  if (!src) return NULL;
  Borrow * copyHead = NULL, * copyTail = NULL;
  while (src)
  {
    Borrow * newNode = createBorrow (src → studentID,
                src → bookID, src → title, src → date);
    if (!copyHead)
    {
      copyHead = copyTail = newNode;
    }
    else
    {
      copyTail → next = newNode;
      copyTail = newNode;
    }
    src = src → next;
  }
  printf ("Borrowing records cloned successfully!\n");
  return copyHead;
}
// Count total books borrowed
void countBooks ()
{
  Borrow * temp = head;
  int count = 0;
```

```
    while (temp)
    {
        count++;
        temp = temp -> next;
    }
    printf("Total Books Borrowed : %d\n", count);
}

// === Main Menu ===
int main()
{
    int choice, studentID, bookID;
    char title[100], date[20];
    Borrow *auditList = NULL;
    while(1)
    {
        printf("\n=== Library Book Borrowing System ===\n");
        printf("1. Issue a Book\n2. Return a book\n
3. Search by student ID\n");
        printf("4. Display All Records\n5. Clone
Records (Audit)\n6. Count Total Books Borrowed\n
7. Exit\n");
        printf("Enter choice: ");
        scanf("%d", &choice);
        getchar(); // clear newline
```

```
switch (choice)
{
  case 1:
    printf("Enter Student ID: ");
    scanf("%d", &studentID);
    printf("Enter Book ID: ");
    scanf("%d", &bookID);
    getchar();
    printf("Enter Book Title: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0;
    printf("Enter Date of Issue: ");
    fgets(date, sizeof(date), stdin);
    date[strcspn(date, "\n")] = 0;
    issueBook(studentID, bookID, title, date);
    break;

  case 2:
    printf("Enter Book ID to return: ");
    scanf("%d", &bookID);
    returnBook(bookID);
    break;

  case 3:
    printf("Enter Student ID: ");
    scanf("%d", &studentID);
    searchByStudent(studentID);
    break;
```

```
        case 4:
            displayAll();
            break;
        case 5:
            auditList = cloneList(head);
            displayAll(auditList);
            break;
        case 6:
            countBooks();
            break;
        case 7:
            printf("Exiting program... Goodbye!\n");
            exit(0);
        default:
            printf("Invalid choice! Try again.\n");
    }
}
return 0;
}.
```

Output Clear

```
===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 1
Enter Student ID: 101
Enter Book ID: 3523
Enter Book Title: programmin in c
Enter Date of Issue: 2025-08-10
Book issued successfully.

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 1
Enter Student ID: 103
```

```
Enter Book ID: 3690
Enter Book Title: data structures
Enter Date of Issue: 2025-08-15
Book issued successfully.

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 1
Enter Student ID: 105
Enter Book ID: 3801
Enter Book Title: fundamentals of java
Enter Date of Issue: 2025-09-20
Book issued successfully.

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
```

```
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 3
Enter Student ID: 105

Student Borrow Records:
StudentID    BookID     Title                          Date
-----------------------------------------------------------------
No records found for Student ID 105.

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 5
Borrowing records cloned successfully.

All Borrowing Records:
StudentID    BookID     Title                          Date
```

```
Borrowing records cloned successfully.

All Borrowing Records:
StudentID    BookID     Title                          Date
-----------------------------------------------------------
102          3510       programmin in c                2025-08-10
3429         3901       fun with python                2025-07-27

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 6
Total Books Borrowed: 2

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
```

```
6. Count Total Books Borrowed
7. Exit
Enter choice: 4

All Borrowing Records:
StudentID    BookID     Title                          Date
-----------------------------------------------------------------
102          3510       programmin in c                2025-08-10
105          3457       Data structures                2025-10-20
3429         3901       fun with python                2025-07-27

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 2
Enter Book ID to return: 3457
Book with ID 3457 returned successfully.

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
```

```
3429          3901        fun with python                 2025-07-27

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 6
Total Books Borrowed: 2

===== Library Book Borrowing System =====
1. Issue a Book
2. Return a Book
3. Search by Student ID
4. Display All Records
5. Clone Records (Audit)
6. Count Total Books Borrowed
7. Exit
Enter choice: 7
Exiting program... Goodbye!


=== Code Execution Successful ===
```