

ALGORITHM:

1. Start
2. Initialize book and member databases
3. Display menu options:
 - 1: Add new book
 - 2: Remove a book
 - 3: Check out a book
 - 4: Return a book
 - 5: Search for a book
 - 6: Generate overdue book report
 - 7: Exit
4. Based on user selection:
 - If 1: Input book details and add to the database
 - If 2: Remove book from database if exists
 - If 3: Check if book is available, assign to a member with a due date
 - If 4: Check return date and calculate late fees (if any)
 - If 5: Search by title, author, or ISBN and display results
 - If 6: Generate and display overdue book report
 - If 7: Exit the program
5. Return to the menu after each operation
6. End

PSEUDOCODE:

BEGIN

Initialize book_database, member_database

DO

DISPLAY "Library Book Management System"

DISPLAY "1. Add Book"

DISPLAY "2. Remove Book"

DISPLAY "3. Check Out Book"

DISPLAY "4. Return Book"

DISPLAY "5. Search Book"

DISPLAY "6. Generate Overdue Report"

DISPLAY "7. Exit"

INPUT choice

IF choice = 1 THEN

 CALL AddBook()

ELSE IF choice = 2 THEN

 CALL RemoveBook()

ELSE IF choice = 3 THEN

 CALL CheckOutBook()

ELSE IF choice = 4 THEN

 CALL ReturnBook()

ELSE IF choice = 5 THEN

 CALL SearchBook()

ELSE IF choice = 6 THEN

 CALL GenerateOverdueReport()

ELSE IF choice = 7 THEN

 EXIT

ELSE

 DISPLAY "Invalid Choice"

ENDIF

WHILE choice != 7

END

PROCEDURE AddBook()

 INPUT book details (title, author, ISBN, status)

 ADD book to book_database

END

```
PROCEDURE RemoveBook()

INPUT book ID

IF book exists THEN

    REMOVE book from book_database

ELSE

    DISPLAY "Book Not Found"

ENDIF

END
```

```
PROCEDURE CheckOutBook()

INPUT book ID and member ID

IF book available THEN

    UPDATE book status to "Checked Out"

    SET due date

ELSE

    DISPLAY "Book Not Available"

ENDIF

END
```

```
PROCEDURE ReturnBook()

INPUT book ID

IF book overdue THEN

    CALCULATE late fee

ENDIF

UPDATE book status to "Available"

END
```

```
PROCEDURE SearchBook()

INPUT search query (title, author, ISBN)

SEARCH book_database

DISPLAY results
```

END

PROCEDURE GenerateOverdueReport()

FIND books past due date

DISPLAY overdue books with late fees

END

FLOWCHART:

