Computer Organization and Assembly Language: Project Proposal

Project Title:

Library Management System in Assembly Language

Proposed to: Mr. Talha Shahid

Prepared By:

Abdul Saboor
i233039
FAST-NUCES

1. Introduction

Managing library operations such as tracking books, student registrations, and handling rentals can be challenging in the absence of an automated system. This project aims to design and develop a **Library Management System** using **Assembly Language (x86, MASM/Irvine32)** to demonstrate how low-level programming can handle real-world data operations and user interaction.

2. Project Objectives

- To build a functional library management system using x86 Assembly Language.
- To gain practical experience in working with data structures, memory management, and user interfaces in assembly.
- To allow users (librarians and customers) to perform operations such as adding books, renting books, and updating records.
- To demonstrate the capability of Assembly Language in creating structured, menu-driven applications.

3. Scope of the Project

The system will include the following functionalities:

For Librarians:

- **Add New Books** to the system.
- **Update Book Information** (title, author, quantity).
- **Display Library Track Info** (list of all books with quantities).
- **Display Book Info** (search by ISBN).

- **Register Students** (record their ID and name).
- **Exit the Program.**

For Customers (Students):

- **Log In** using their Student ID.
- **Rent a Book** (only one book at a time per student).
- **Return a Book** (and increment the available quantity).
- **Log Out** of the system.

4. System Requirements

- **Software:**
  - MASM (Microsoft Macro Assembler)
  - Irvine32 library
  - Windows OS with 32-bit support
- **Hardware:**
  - Intel x86 processor
  - Minimum 2GB RAM
  - Standard I/O device (keyboard for input, screen for display)

5. Functional Description

Main Components:

- **Welcome Interface:**
  Displays stylized welcome text and developer attribution using colored ASCII art.

- **User Role Selection:**
  Prompts user to choose between "Librarian" and "Customer".

- **Book Management:**
  Books are stored with metadata: Title, Author, ISBN, Quantity. They can be added, updated, or displayed.

- **Student Management:**
  Students are registered by ID and name. Each student is permitted to rent only one book at a time.

- **Book Renting System:**
  When a student rents a book:

  - The system checks availability.
  - If available, quantity is decremented.
  - Book is assigned to the student.

- On return:

  - The system matches the book by ISBN.
  - Increments the book quantity.

- **Storage Structures:**

  - Arrays for up to **50 books** and **50 students**.

    - Uses indexed memory for data access (titles, authors, ISBNs, etc.)

6. Technical Challenges

- Manual string manipulation and memory management.
- Simulating data structures (like arrays) in a language without native support.
- Implementing logic-heavy features (like renting/returning books) using low-level conditional and loop operations.

7. Expected Outcomes

- A working command-line interface that:
    - Allows dynamic data entry and retrieval.
    - Tracks book availability and student-book relationships.
- A demonstration of structured programming principles in Assembly.
- Source code with modular procedures for maintainability and readability.

8. Conclusion

This project bridges the gap between low-level programming and high-level system design. It not only reinforces knowledge of Assembly but also reflects how fundamental operations in modern software (like library systems) are built from the ground up.

9. References

- Kip Irvine, *Assembly Language for x86 Processors*

- MASM Programming Guide

- Irvine32 Library Documentation