

Project 1 – Library Management System

CS 251, Fall 2023

Collaboration Policy

By submitting this assignment, you are acknowledging you have read the collaboration policy in the syllabus for projects. This project should be done individually. You may not receive any assistance from anyone outside of the CS 251 teaching staff.

Late Policy

You are given a grace period of 24-hours. You do not need to let anyone know you are using this grace period, you simply continue to work and submit during the time period. Beyond this period, no late submissions will be accepted. Please note that if you are continually using the grace period, then you are off track from our anticipated pace that facilitates spacing of assessments.

What & Where to Submit

1. zyBooks - library.cpp code for functionality score
2. Gradescope - library.cpp for manual grading

Do not submit any additional files.

Table of Contents

[Project Summary](#)

[The Data](#)

[Various "Error" Messages](#)

[Executing and Using the Solution](#)

[Milestone 0 - Starter Files & Strategies](#)

[Milestone 1 - Data Structures & Exiting](#)

[Milestone 2 - Add Entry to the Library](#)

[Milestone 3 - Display the Library](#)

[Milestone 4 - Clear the Library](#)

[Milestone 5 - Load Library From File](#)

[Milestone 6 - Output Library To A File](#)

[Milestone 7 - Print the Checkout Stats](#)

[Milestone 8 - Search the Library](#)

[Milestone 9 - Remove an Entry](#)

[Milestone 10 - Print Out Invalid Entries](#)

[Grading Breakdown](#)

Copyright Notice

Copyright 2023 Adam T Koehler, PhD - University of Illinois Chicago

This assignment description is protected by [U.S. copyright law](#). Reproduction and distribution of this work, including posting or sharing through any medium, such as to websites like [chegg.com](#) is explicitly prohibited by law and also violates [UIC's Student Disciplinary Policy](#) (A2-c. Unauthorized Collaboration; and A2-e3. Participation in Academically Dishonest Activities: Material Distribution).

Material posted on [any third party](#) sites in violation of this copyright and the website terms will be removed. Your user information will be released to the author.

Project 1 – Library Management System

CS 251, Fall 2023



Project Summary

A library curates and stores potentially thousands of books. To concentrate on specific topics, in this project we will implement a library management system utilizing a simplified data set compared to all the information normally stored by a library's systems. In addition to simplifying the data, we have a specific set of instructions that our management system can perform.

Program and Coding Restrictions

- Any hard coding, especially to pass test cases, will result in a zero for the entire project.
- You cannot use any libraries beyond the ones we have given.
- You must use output streams, file streams, vectors, and strings.
 - You are not allowed to use any other containers, objects, or global variables.
- You may not use structs, pointers, or dynamic memory allocations.
- Do not change any of the lines we have already coded for you in the given code.

Primary Project Topics

- String Manipulation
- File IO
- Vectors

Given Files (3)

We have provided a single C++ file as well as two CSV files. The C++ file is for your implementation. The database.csv file provides an initial database file for your program to work with. The database_output.csv file provides an export from within the program of the initial database that you can utilize for comparison to your export. You can generate additional database files following similar patterns to the provided example.

Library Management System Interactions

You will implement each of the system's functionalities.

- | | |
|--|----------------------------------|
| • A - Add Item To Library | • O - Output Library To File |
| • C - Clear The Library Of All Entries | • P - Print Out Checkout Stats |
| • D - Display Library Entries | • R - Remove A Library Entry |
| • I - List Invalid Library Entries | • S - Search For A Library Entry |
| • L - Load Library From File | • X - Exit Program |

Project 1 – Library Management System

CS 251, Fall 2023

The Data

As seen in the database.csv file, there are three items tracked by the library management system. The book title, a 13-digit ISBN number, and a checkout status for the library entry.

Where the Wild Things Are, 978-1-43-520892-6, 2

Book Title

Cannot Contain Commas
Can Be One or More Words

ISBN Number

13 hyphen separated digits

Default Value:
000-0-00-000000-0

Checkout Status

Integer representation of status.

0 - In Library (default value)

1 - Checked Out

2 - On Loan

3 - Unknown State

All Others - Invalid State

Various "Error" Messages

The following are the error messages produced in various places throughout the program.

- Unknown command.
- The book title cannot contain commas.
- Could not find the database file.
- Could not open database file for writing.
- Invalid remove by choice option.
- Invalid search by choice option.
- The library has no books.
- The library has no invalid entries.

Executing and Using the Solution

We have provided write-ups for each system functionality. However some descriptions are shorter with the expectation that you will execute the solution to further clarify functionality. The executable solution can be found within zyBooks.

Milestone 0 - Starter Files & Strategies

Completely read this entire project description and all the provided [starter files](#). If you need a starting point reference for implementing, I suggest (re)watching the week 2 supplemental video [C++ Coding: File Streams, Vectors, and Input Redirection](#).

You should only use `getline()` to acquire user input in this project. This will avoid issues when mixing `getline` and the extraction operator. To help match output, every `getline` that acquires typed input (uses `cin` as the stream) should be immediately followed by outputting a newline.

You should utilize the [character checking predicate functions in the cctype library](#) to achieve part of your algorithm, notably the [isspace](#) and [isdigit](#) functions but others may also be useful.

We strongly suggest using function decomposition as it will save you headaches when debugging your code, and it will be part of the manual style grade. Additionally, within each milestone you should be thoroughly testing all prior and newly added functionality.

Project 1 – Library Management System

CS 251, Fall 2023

For each functionality implementation:

Execute the solution to demonstrate the functionality to yourself including outputs and inputs. Preciseness will matter as zyBooks does a 1 to 1 output comparison similar to the other exercises you have completed. When necessary, you can copy items from the provided database file or type in your own. Before you code, outline your algorithm and the needs of the functionality as well as what you expect the command to do from the menu prompt and descriptions in this document.

Solution execution can also help identify what your program should do with errant input(s).

Milestone 1 - Data Structures & Exiting

Make sure you can successfully compile, execute, and exit the program. Now, within the main function, create the data structures necessary to store each of the data pieces for every library entry. Remember that you will use `getline()` for all user input from the console.

Milestone 2 - Add Entry to the Library

Add an individual entry into the library by acquiring a book title and ISBN number. Remember, book titles cannot contain commas. Here is an example you can use:

Book Title: Where the Wild Things Are
ISBN: 978-1-43-520892-6

Milestone 3 - Display the Library

Output the entries from the library to the console. For example this library has one entry:

Your Current Library

Where the Wild Things Are --- 978-1-43-520892-6 --- In Library

Milestone 4 - Clear the Library

Clear the data structures utilized to store the library entries, resetting the library to a start state.

Milestone 5 - Load Library From File

Using a comma separated data file (similar to the one provided), load the new entries from a provided data file. Note that the internal library entries individual data pieces should not have commas, leading whitespace, or trailing whitespace so these issues should be resolved when parsing each entry before storing the entry's data into the library. [Default values](#) should be used when lines do not have a value, values will always be in the file in the expected order. See the given file for some examples of lines with missing information.

Milestone 6 - Output Library To A File

Output each entry to a file separating the data pieces by commas to produce a file similar to the output datafile provided as an example. The example output file is an execution of the command immediately after reading in the provided database into the program's library.

Project 1 – Library Management System

CS 251, Fall 2023

Milestone 7 - Print the Checkout Stats

Output the checkout stats for the library entries to the console. Stats should include total books, the quantity of books in the library currently, the quantity of books checked out, the quantity of books on loan, the quantity of books with the unknown state value, and the quantity of books containing all other state values.

| | |
|------------------------------|---|
| Your Current Library's Stats | |
| ----- | |
| Total Books: | 0 |
| In Library: | 0 |
| Checked Out: | 0 |
| On Loan: | 0 |
| Unknown: | 0 |
| Other: | 0 |

Milestone 8 - Search the Library

Based on user input, perform a search on either the names or ISBN numbers to discover the last library entry to match the user input.

Milestone 9 - Remove an Entry

Based on user input, perform a search on either the names or ISBN numbers to discover and remove the last library entry to match the user input.

Milestone 10 - Print Out Invalid Entries

Search for and output (to the console) each invalid entry in the library. The book title is never invalid, the ISBN can be invalid in two ways, and the checkout state is invalid if the numerical representation is not 0, 1, 2, or 3. The 13-digit ISBN number is invalid if it does not have 13-digits or if the first 12 digits do not properly calculate the check digit's value using the [ISBN-13 check digit calculation algorithm](#).

The algorithm calculates the weighted sum of the 12 digits, where the weight is either 1 or 3 depending on the digit. For example a number 426 would have a weighted sum of:

$$4 * 1 + 2 * 3 + 6 * 1 \text{ or a summed total of } 16$$

After calculating the weighted sum, the remainder when dividing by 10 is calculated. Then the remainder is subtracted from 10. The result of the subtraction is the check-digit value, except if the result of this subtraction is 10 the check digit is 0.

Requirements Reminders

1. You cannot use any libraries beyond the ones we have given.
2. You cannot change the provided code.
3. Each input file may be opened and read exactly once for each load call.
4. Your code file must have a header comment with your name and a program overview. Any functions you have decided to implement must have a header comment above the function, explaining the function's purpose, parameters, and return value (if any). Inline comments should be supplied as appropriate; comments like "declares variable" or "increments counter" are useless. Comments that explain non-obvious assumptions or behavior are appropriate.

Project 1 – Library Management System

CS 251, Fall 2023

5. No global variables; use parameter passing and function return. No structs. No classes. No pointers. No heap allocation. No use of containers except for vectors.
6. The cyclomatic complexity (CCN) of any function should be minimized. In short, cyclomatic complexity is a representation of code complexity — e.g. nesting of loops, nesting of if-statements, etc. You should strive to keep code as simple as possible, which generally means encapsulating complexity within functions (and calling those functions) instead of explicitly nesting code.

As a general principle, if we see code that has more than 2 to 3 levels of explicit looping that score will receive grade penalties. The solution is to move one or more loops into a function, and call the function.

Don't forget to submit your code to zyBooks for functionality testing and submit your final code to Gradescope! Failure to submit to Gradescope will result in a zero on the project.

Grading Breakdown

Penalties may occur when manually grading for things like hard-coding or code restriction violations, but here is the breakdown for positive point acquisition.

| | | |
|--------------------------------|--|-----------|
| Functionality Score | Milestone 2 - Add Entry to the Library | 8 |
| | Milestone 3 - Display the Library | 3 |
| | Milestone 4 - Clear the Library | 3 |
| | Milestone 5 - Load Library From File | 13 |
| | Milestone 6 - Output Library To A File | 6 |
| | Milestone 7 - Print the Checkout Stats | 6 |
| | Milestone 8 - Search the Library | 7 |
| | Milestone 9 - Remove an Entry | 8 |
| | Milestone 10 - Print Out Invalid Entries | 6 |
| | Sub Total | 60 |
| Manual Score | Function Decomposition | 5 |
| | Style and Other Items | 5 |
| | Sub Total | 10 |
| | Total | 70 |

Project 1 – Library Management System

CS 251, Fall 2023

One Example Execution

The menu print out has been condensed as it is part of the given code.

Input

```
D
L
database.csv
d
X
```

The above input can be typed or inserted into a file. It produces the output to the right when using input redirection with an input file.

```
./a.out < input.txt
```

Output

```
Welcome to the Library Management System
-----

Library Management Menu
-----
A - Add Item To Library ...
X - Exit Program

Enter a command (case does not matter):
Your Current Library
-----
The library has no books.

Library Management Menu
-----
A - Add Item To Library ...
X - Exit Program

Enter a command (case does not matter):
What database to read from?
Read in 11 lines from the file.

Library Management Menu
-----
A - Add Item To Library ...
X - Exit Program

Enter a command (case does not matter):
Your Current Library
-----
The Hunger Games --- 978-1-40-710908-4 --- In Library
Red Rising --- 978-0-55-339079-7 --- In Library
Where the Wild Things Are --- 978-1-43-520892-6 --- On Loan
Goodnight Moon --- 978-0-06-020706-9 --- Checked Out
The Ultimate Hitchhiker's Guide --- 978-3-89-687070-4 --- In Library
Star Wars: Ahsoka --- 978-1-40-528790-9 --- In Library
Book --- 000-0-00-000000-0 --- In Library
Master & Apprentice --- 978-0-52-561937-7 --- Checked Out
The C Programming Language --- 978-0-87-692596-6 --- Invalid State
Oops --- 978-12-40-528790-9 --- In Library
Snow Crash --- 978-3-44-223686-2 --- Unknown State

Library Management Menu
-----
A - Add Item To Library ...
X - Exit Program

Enter a command (case does not matter):
```