Taras Glushko
CSCE 1040.001
Homework 3

Library Management System Report

Introduction

This project for a Library Management System was developed from the initial design provided in Homework 2. This system is designed to manage basic library functionalities, including addition and management of books, patron registration, and borrowing/returning of books by patrons. The report will be an overview of the general structure and system design, class design, implementation, and interaction of different components of the system. This also illustrates how everything changed from the initial draft in Homework 2 to the final implementation in this submission.

The project was implemented in C++ using the rules of object-oriented programming. This made the structure modular and maintainable. The mastering of STL collections lead to the use of these collections in effectively managing various sets of data, such as books, patrons, and loans. In sum, the system is constituted by six main classes and some extra functionality concerning loan management and fines, which allows for further extensibility on the fly.

Overview of Classes

This LMS includes six significant classes, each of which is responsible for another kind of functionality of the system.

The Patron class will maintain information with respect to library patrons, including but not limited to the name of the patron, their ID, fine balance, and number of books checked out. It will also include functions for updating patron information, addition of books to their record, paying their fines, and returning books. The Book class provides all information relevant to a book: title, author, ISBN number, library ID, its status, and cost. It does not contain methods that can change the state of the book; in other words, to be borrowed or returned.

The Loan class is at the heart of the system because it associates a patron with a book being lent. This will maintain a record of the loan ID, who borrowed the book, the book loaned, and its due date. It also allows methods for setting and updating the due date for loans. The Library class acts as the overall controller for the system, responsible for general operations across the library. This system deals with adding

new books and patrons into the system, borrowing and returning of the books through the loan process, and listing of overdue books.

Though it was not implemented, there is room in this design for a Fine Manager class that would determine fines for late or lost books, and ensure the patrons were charged accordingly. An Inventory class could be employed to manage records of the collections of books, patrons, and loans to make querying and updating of those records easier.

Design Changes from Homework 2

Incorporated within this final design were several key improvements and changes from the original design submitted in Homework 2. Perhaps one of the most important edits made within this design was the complete integration of the loan system. The loan system properly checks to prevent overflow of the borrowing limit of patrons, and it blocks books from being borrowed whenever they are checked out or marked as lost. Another critical enhancement was the handling of the status of the books. It can now trace the status of the books in cases where books borrow, return, or get marked lost.

While a fines system was partially implemented, the design was enhanced so that in the future, extensions could more fully calculate and track fines for overdue or lost books. Also added was a Makefile, which eased compilation of the project by making sure all the files are compiled together without having to manually intervene. These aspects ensure the robustness, modularity and expandability of the system.

Challenges and Considerations

One of the difficulties with this project has been guaranteeing that classes interact seamlessly: Patron, Book, and Loan. This involves a lot of care in managing pointers and references to avoid memory management-related problems. Another challenge will be in providing an altogether robust system for tracking the status of books and making sure that patrons do not exceed the number of allowed books they borrow. This included requiring a system to validate that a patron has no overdue books, nor reached the borrowing limit, before further checkout was allowed.

The loan process was highly defined in managing this interaction of the books correctly to the patrons and updating their respective status upon return. The overall design and modular structure also allow easy maintenance and extensibility of the system.

Conclusion

Homework 2 was utilized in the development of the Library Management System, where design improvements and refinements were applied. The project uses object-oriented principles in managing the lending of books, patron registration, and fines. The present design extends this to accommodate future expansion in certain areas: fine computations with more comprehensiveness, and reporting capability. It is a good backbone for the management of library operations. The system is modular, and hence, it is maintainable with the possibility of extending it for further development.