Vg101. Introduction to Computer and Programming (Fall 2019)

Homework #7

Assigned: 11 NOV 2019 Due: 25 NOV 2019

Notes for submission: We are going to use the online judge (OJ) to grade your homework, so please submit your homework in the OJ. For backup purpose, you should also submit your homework to the Canvas. **For canvas submission, please name the file using the format bellow: "[Your last name][Initial of your first name]_sYourStudentID_hw7.zip" which includes all your source codes. For example, WuJ_s123456789_hw7.zip**. And the source file name for each problem should be Problem1.c, Problem2.c, etc.

All of the following programs should be written in C

1. Implement a simple library management system. The system has a user interface to accept user command (with a prompt ">") and respond accordingly. To make your life easier, let's assume that the number of customers <= 100, and the number of books <= 1000. The following are the user commands that can be accepted:

1) **addbook** *bookID book_title*
Add a new book to the library given the book ID number and the book title. You should check if the bookID number already exists in the system. If it is, let the user know. Here let's assume that the book_title is a character string without the space characters.

2) **addcustomer** *customerID customer_name*
Add a new customer of the library, given the customer ID number and the name. You should check if the customerID number already exists in the system. If it is, let the user know. Here let's assume that the customer_name is a character string without the space characters.

3) **checkb1** *bookID*
List the information for the given book ID. The book information includes ID, title, and status ("borrowed" or "on shelf").

4) **checkb2** *book_title*
List the information for the given book title. The book information includes ID, title, and status ("borrowed" or "on shelf"). Note that different books may use the same name.

5) **checkc** *custormerID*
List all the books borrowed by the customer, given the ID number.

6) **borrow** *bookdID customerID*
Perform the book borrow action. If the book was already borrowed, you should prompt for the information.

7) **return** *bookID*
Perform the book return action. Of course, you cannot return a book that has not been borrowed.

8) **listbook**
List all the books in the system.

9) **listcustomer**
List all the customers in the system. The customer information includes ID and name.

10) **exit**
Exit the program

If the user type any other commands, just prompt for "unknown command". You should design the underlying data structure to support the above commands. A sample run of the program is shown below, **please follow the output format in the sample**:

Welcome to the library management system, please type your commands.
> listbook
There is no book in the system.
> listcustomer
There is no customer in the system.
> addbook 1 book1
book1 added to the system.
> addbook 1 book2
The book ID already exists.
> addbook 2 book2
book2 added to the system.
> checkb1 1
1 book1 on shelf
> addcustomer 1 name1
name1 added to the system.
> addcustomer 1 name2
The customer ID already exists.
> addcustomer 2 name2
name2 added to the system.
> borrow 1 1
Success! book1 is borrowed by name1.
> checkb1 1
1 book1 borrowed
> borrow 1 2
Fail! The book was borrowed by another customer.
> listbook
1 book1 borrowed
2 book2 on shelf
> checkc 1
Book borrowed by name1:

1 book1
> return 1
Success! The following book has been returned:
1 book1
> return 2
Fail! The book has not been borrowed.
> aloha
Unknown command!
> exit
Exit the system!