# Algorithms and Data Structures 2

## Assignment 1
## This assignment will be marked out of 100% but is worth 15% of the module

A library wishes to keep a record of all the books that it has in stock. You need to create a system in C that displays a menu which allows the following options to be presented to the user:

1. Add a new book to the library
2. Allow customer to take out a book
3. Allow customer to return a book
4. Delete an old book from stock
5. View all books
6. View a specific book
7. View details of most popular and least popular books
8. *This should be an appropriate option that you provide*
9. Exit the system

The underlying data structure used in this assignment will be a **linked list.**

The details of a book are as follows :

- book identifier (8 digit number in the form xxxx-xxxx),
- book name,
- book author,
- year of publication (cannot be older than the year 2008)
- a Boolean value that indicates that status of the book i.e. if the book is currently loaned out or not.
- Customer name – this will be set to a customer name if the book has been loaned out, otherwise it is an empty string.
- Number of times the book has been loaned out, this will be initialised to 0 for new books
- *You may add an extra feature for a book here if it is relevant to menu option 8.*

Your program will also contain the ability to store the data in a file. Therefore, when your program begins executing, the following actions should take place:

- The program should check to see if file `book.dat` exists on your drive.

- If it doesn't then a message should appear on the screen to indicate to the user that a database of books doesn't exist and so the user will need to input the books.

- If the file does exist, then the books in the file should be copied into the linked list. A message should appear to say that the system has been populated with books from the data file.

- The menu should then appear and the user should be asked to input an option. If the user chooses a valid option (1 - 9), this option will execute. Choosing an invalid option will result in an error message.

- Every time a menu option has competed executing, the menu should reappear to allow the user choose another option.

- Once the user indicates that they wish to exit the system (by choosing option 9), the system should indicate that it is going to copy all of the books back to the database (i.e. the book.dat file).

Specific details of some of the menu options are as follows :

- Add a book
  - Assume that there is a maximum of 10 books in the library. An error message will appear if the list is full. Ensure that all book identifiers are unique and that the date of publication is not older than the year 2008.

- Delete a specific book
  - This option is to remove old books from the library stock. The user will need to indicate which book they wish to delete by inputting a book identifier. Note that only books older than 2010 can be deleted.

- Take out a book
  - The identifier of the book is input. If the book is found and it's not already loaned out, then the customer name is input and added to the book, also the status of the book is updated and the value storing the number of times the book has been taken out is updated by 1. If the book isn't found or has already been loaned out then appropriate error message should be displayed.

- Return a book
  - The identifier of the book and customer name are input. If the book is found and it has been taken out by that customer then its status is updated accordingly otherwise appropriate error messages are displayed.

- Judging the popularity of books is based on the number of times the book has been on loan.

**Notes:**

- Your program should contain functions. Examples of parameter passing (passing by reference and passing by value) should be present in some of your functions.

- Your program should be user-friendly by displaying messages informing the user what is happening. It should also contain error checking and present useful error messages to the users if they input an incorrect value.

- Your program should be well structured and should contain lots of comments to explain the logic behind your code.

- Projects **MUST** be your own work. Students will be penalised for plagiarism.

- Save your program in a file called ads2_assign1.c and submit it via Brightspace by 6pm on Sunday 28th March 2021.