# Assignment

## Assignment Description

Write an application in C/C++ to process library book records. The library book records should be read from a text file containing the following data: *ISBN*, *author*, *title*, *edition*, *publisher*/*imprint* and *price*. Use the Internet or UTAR library OPAC to gather data on at least 20 records and prepare them in the text file. As each book record is read, insert it into a list by using ***array of structures***.

After building the list, display a menu with the following options:

(a) **List** – Display the list in tabular format. Each display should contain an appropriate heading and column captions;

(b) **Search** – Search for book record(s) using the ***ISBN***, ***author*** or ***title***, and print the full record for the book(s). It can match word(s)/number within a string/number, and also print multiple results. This is best explained by the following examples:

- If the book titles "**Introduction to Java Programming**", "**Java: How to program**", and "**Big Java**" are in the list, then searching for "**Java**" must print all three records (because the word "**Java**" occurs in all three book titles).
- If the author name "**Paul J. Deitel**" is stored inside the list, searching for either "**Paul**" or "**Deitel**" should return the record.

Note that you may need to use `strstr()` which allows you to locate substring.;

(c) **Delete** – Delete an existing book record from the list;

(d) **Exit** – Stop the program.

**Advance Questions: Maintaining book records – duplicate or multiple copies**

The library purchases multiple copies or duplicate books of the same book. The duplicate books might be in a range of different editions contain new, deleted or revised information. Modify the storage structures of book records and implementations in your program, so that it can store more details of books, such as number of editions, number of copies, location of each copy, status of each copy, etc.

**Additional Information**

To make your program more robust and avoid problems at run time, do as much status/error checking as you could in your program. You may also add on more features in your program for enhancement.

## Assessment and Submission

This is a group assignment. Form a group of 2 or 3 members, preferably from same programme as yours. Prepare a report (preferable using word processing software) to answer the questions given above.

Your **REPORT SHOULD CONTAIN** the following:

1. design of the application (structure chart and flowcharts/pseudocode)
2. print out of the C++ program.
3. sample output(s) (Alt+Prnt Scrn) of your program.
4. sample of input data and test cases
5. soft copy of the C++ program and input text file (upload to WBLE)

Do remember to print the assignment marking sheet and attach as the FIRST PAGE of your report.

This practical assignment will contribute 20% of your final mark. Refer to the marking sheet for the mark allocations for the report and C++ program. The report will be marked for *correctness*, *completeness*, *presentation style*, and *relevant use of diagrams/tables/graphs*, etc. And the C++ program will be marked for *correctness*, *completeness*, *program style*, *adequate testing* and *documentation/comments*. It's your responsibility to understand the requirements of the tasks and prepare well for your submission. You will not receive full mark if you do not submit the report and C++ program that is a reasonable attempt and compiles without error. You might be asked questions about the works you submit to ensure that you understand them.

## Plagiarism

It is important that your solutions to the practical assignment be your own work. It is perfectly acceptable to seek help and advice when completing the practical assignment, but this must not be taken to the point where what is submitted is in part someone else's work.