

## CA Exercise 2 – Book and Character Information System

---

The objective of this CA exercise is to create an application for storing and retrieving information on books and the characters that feature in them.

The application should allow/support the following:

- The addition of new books to the system.
  - Title, author, year of publication, publisher, number of pages, genre (fantasy, thriller, etc.), plot description, book cover image (URL), and so on.
- The addition of new characters to the system.
  - Name, gender, description, etc.
- Ability to associate characters with books they feature in (e.g. Robert Langdon features in the Da Vinci Code, Angels and Demons, Inferno, and a number of other Dan Brown books).
- Ability to edit/update/delete books and characters, etc.
- Ability to search for books and characters (by name, by author, by genre, by year, etc. as appropriate). The key thing is that a search facility is provided with some search options.
- Listings of search results should be appropriately sorted depending on the search parameters/options (e.g. alphabetically and/or by year and/or by genre). The key thing is that the user should have some options on how to sort the search results.
- The system should support some level of interactive “drill down” for additional detail. For instance, searching for books by author Dan Brown should provide a list of his books; one book could be clicked on to see more information specifically on that book; that detail could include a list of characters in the book; clicking on a character opens up detail on the character; details on the character includes a list of books they have featured in; this list of books could be clicked on to see more information; and so on.

### Notes

- This is a team CA exercise. Teams consist of 2 students only. Students should find/choose their own partners.

- You essentially have to implement the required data structures and algorithms from scratch and from first principles (in line with the module learning outcomes).
  - You are free to reuse any of your generic code from CA Exercise 1.
- You have to use hashing as part of the project to avoid (excessive) linear/sequential searching. For instance, a custom hash function should be used for retrieving details on a specific character or book.
- Remember that the key point of this CA exercise is to demonstrate knowledge and proficiency with hashing and sorting in particular. Keep this in mind!

### **Indicative Marking Scheme**

- Create/add facilities (book and character) = 15%
- Edit/update/delete facilities (books and characters) = 15%
- Search and listing facilities (books and characters) = 15%
- Sorting of search results/listings (books and characters) = 10%
- Hashing for individual search/lookup (books and characters) = 10%
- Persistence facility (saving and loading data) = 10%
- Graphical user interface = 10%
- General (commenting, style, logical approach, completeness, etc.) = 15%