1 Brief Task Description

You must design and implement a library search system allowing users to search for books by title. It should also allow adding new books and removing lost/damaged books from the library system. As the library will have millions of books, you should think about the most appropriate data structure(s) and algorithms for the search; add and remove functionality. During the design phase, you should analyse the algorithms for searching for a book, adding a book and removing a book from the system. You should not use any third party libraries or code as a part of your solution and all code should be written by you, i.e. not automatically generated. You also shouldn't use non-standard or operating system dependant libraries in your program (the code should all be standard C++). To achieve a good mark, you should design and implement the data structures yourself. Using standard template library (STL) data structures will severely limit the grade you can achieve.

2 Submission

You must submit a single zip file of all required source code; a single PDF report and a video demonstration of the software by Friday, 11th April 2021. The source code zip file should include: • C++ source code files of your program

• catch2 test code

• a Makefile to compile your program The report must be no longer than 8 pages (including references), with a font size of 11 or 12 point. If your report exceeds 8 pages only the first 8 pages will be marked. The video demonstration should be very short and is not a presentation, it should clearly demonstrate your knowledge of the implementation. There is a strict 5 minute time limit and only the first 5 minutes of any longer videos will be used in marking your work. Note:

• Only source code (header and cpp files) and a Makefile should be included in the zip file, no other files

• If code does not compile and run it will severely limit your marks for the code

• As anonymous marking will be applied, you should not include your name in your source code or report •

A penalty will be applied to the mark for submissions not meeting these requirements, e.g. name, type and contents of submission

1 • Marks for code will be severely limited if you cannot demonstrate understanding in the demonstration video.

3 Scenario A library has found their book search system has become slower as their collection of books has grown. They need you to design an efficient program which allows searching for books by title and adding and removing books. You will be provided with a sample of book data in a text file, which must be loaded by your program. It should, however, be possible for your program to load data from another file, e.g., by using a command line argument of the file name, to allow them to later load the full list of all books in the collection.

4 Detailed Description It is recommended that you complete the tasks in the following order as the later sub-tasks will require the earlier ones.

4.1 Set up Project Create a Git repository and makefile for the project, remember to update the makefile and commit new files to the git repository as you implement/update them.

4.2 Plan Software Design and Implement a Book class to hold the book objects, each book record will have:

- title
- author(s)
- ISBN
- qty available

4.3 Design and Implement Data Structure/Algorithm Design and implement the data structure which will store the Book objects and associated algorithm(s). Analyse the time complexity of the algorithms (this will be needed in your report).

4.4 Test Data Structure Apply software testing to ensure your book class and data structure and algorithms are all working correctly.

4.5 Read Sample Data File Implement the program to read the sample data from the given file and load it into the data structure. 4.6 Implement Menu Implement a (command line) user interface to allow users to search for books by title; add books and remove books. When there are no copies of a given book in the system, the book object should be removed from the system.