Development of a Software Application for a Community Library to Manage Its Movie DVDs

Due Date: 8 June 2022

Weighting: 40%

Group or Individual: A group of 2-3 students (preferably 3 students)

1. Introduction

In this project you develop a software application for a community library to manage its movie DVDs. In the development of this software application, you need to use some data structures and algorithms that are covered in this unit to store, manage, and manipulate the data in the software application. You also need to design algorithms to solve some computational problems arising from the software application and to analyse the time efficiency of your algorithms.

2. Background

You are hired to develop a software system for a community library to manage its movie DVDs. The community library lends movie DVDs to its members. The information about a movie includes title, genre, classification, and duration in minutes. The genre of a movie may be Action, Comedy, History, Drama, or Western. A movie is classified as General (G), Parental Guidance (PG), Mature (M), or Mature Accompanied (MA15+). The community library may have multiple DVDs of the same movie. The software system should keep the information about all the movies in the community library.

To borrow movie DVDs from the community library, one must register with the community library. When a person is being registered with the community library through a staff member, his/her first name, last name, and a contact phone number should be recorded in the software system. The software system should also keep the information about the movie DVDs that are currently being borrowed by each of the registered members.

When a registered member borrows a movie DVD from the library, the movie should be added to the member's borrowing record and when the member returns the movie DVD to the library, the movie should be removed from the member's borrowing record. A registered member can hold up to five (5) movies at any time. However, none is allowed to hold more than one DVD of the same movie at a time.

It is assumed that both the names of registered members and the titles of movie DVDs are unique. In addition, it is assumed that all the DVDs of a movie are identical.

In Phases 1 and 2, you have implemented four (4) ADTs, *Member, MemberCollection*, *Movie* and *MovieCollection*. In this phase (Phase 3), your major task is to use these ADTs to build a software application that satisfies a set of functional and non-functional requirements, which will be given in the next section. In addition, you need to develop an efficient algorithm to solve a computational problem arising from this software development and to analyse the time efficiency

of the algorithm using the empirical algorithm analysis techniques introduced in Lecture 2. Finally, you need to write a technical report.

3. Detailed Tasks

A. Make use of the four (4) ADTs that you implemented in Phases 1 and 2 to build a software application for the community library to manage the movie DVDs, satisfying the following functional and non-functional requirements:

- The software application must be a Console Application of Microsoft Visual Studio 2019/2022 Community Edition on Microsoft Windows 10 and the programming language used in the software development must be C# only.
- The staff members and all the registered members can use the software application. The software application has a main menu and two sub-menus. The identity of the user must be verified in the main menu before he/she can use the functionalities of the system. It is assumed that the username and the password for the staff are 'staff' and 'today123', respectively. Registered members are verified using their first name, last name and a password.
- The software application should allow a staff member to do the following:
  - Add new DVDs of a movie to the system. If the movie is new (the library currently does not any DVD of this movie), then all the information about the movie and the number of the new movie DVDs should be entered into the system; If the movie is not new (the library has some DVDs of this movie), then only the total quantity of the movie DVDs needs to be updated, but the information about the movie needs not to be re-entered.
  - Remove DVDs of a movie from the system. If all the DVDs of the movie are removed, the movie should also be removed from the system.
  - Register a new member with the system. When a member is being registered via a staff member, the member's fist name, last name, contact phone number are recorded in the system, a password is set by the member via the staff. The system must check and make sure the phone number and the password are valid. A contact phone number is valid if it has 10 digits, and the first digit is a '0'. A password is valid if it has 4-6 digits.
  - Remove a registered member from the system. A registered member cannot be removed if he/she has any movie DVD on loan currently.
  - Display a member's contact phone number, given the member's full name.
  - Display all the members who are currently renting a particular movie
- The software application should allow a registered member to do the following:
  - Display the information about all the movie DVDs in alphabetical/dictionary/lexicographic order of the movie title, including the number of the movie DVDs currently in the community library
  - Display the information about a movie, given the title of the movie
  - Borrow a movie DVD from the community library
  - Return a movie DVD to the community library
  - List current movies that are currently borrowed by the registered member

- Display the top three (3) most frequently borrowed movies by the members in the descending order of their frequency. The display should include the title of the movie and frequency (the number of times that the movie has been borrowed by registered members by now)

- The system should have a main menu and two sub-menus similar to those in the following screenshots:

```
================================================================
Welcome to Community Librart Movie DVD Management System
================================================================

=========================Main Menu=========================

1.Staff Login
2.Member Login
0.Exit

Enter your choice ==> (1/2/0)
```

```
=========================Staff Menu=========================

1. Add new DVDs of a new movie to the system
2. Remove DVDs of a movie from the system
3. Register a new member with the system
4. Remove a registered member from the system
5. Display a member's contact phone number, given the member's name
6. Display all members who are currently renting a particular movie
0. Return to the main menu


Enter your choice ==> (1/2/3/4/5/6/0)
```

```
=========================Member Menu=========================

1. Browse all the movies
2. Display all the information about a movie, given the title of the movie
3. Borrow a movie DVD
4. Return a movie DVD
5. List current borrowing movies
6. Display the top 3 movies rented by the members
0. Return to the main menu


Enter your choice ==>(1/2/3/4/5/6/0)
```

B. Design an algorithm and analyse the time efficiency the algorithm using the empirical algorithm analysis technique introduced in Topic 2 for the following computational problem:

- Given an array where the elements are randomly stored, find the three largest elements in the array.

C. Write a technical report, which is organised as below:

- Table of contents
- Introduction – briefly talk about the project and the software application that you have developed
- Design and Analysis of Algorithms (Task B) – Use the pseudocode notations introduced in Topic 1 to describe your algorithm, analyse the time efficiency of your algorithm using the empirical algorithm analysis technique introduced in Topic 2, and apply your algorithm to solve the following computational problem in the software application development:
  - Display the top three (3) most frequently borrowed movies by the members in the descending order of their frequency.
- Software test Plan and Test Results – design a test plan for each of the application's functions. In each of the test plans, please include the test scenarios/cases, actual test data for each of the test scenarios/cases. Please also provide screenshots of the tests and test results.

4. Assignment Requirements

- In this assignment, you are expected to work in a group of 2-3 students, preferably 3 students, and every member in a group is expected to make roughly equal contribution to all aspects of the project.
- It is your own responsibility to find assignment partners. But we will create an channel in the CAB301 Slack to help you find assignment partners. Once you have formed a group, please register your group in the CAB301 Blackboard. To register your group, you click on "Join a CAB301 Assessment Group" in "Assessment Task 3" of the blackboard and follow the instructions there.
- The software application must be a Console Application of Microsoft Visual Studio 2019/2022 Community Edition and the programming language must be C#.
- You must use *MemberCollection* as a data structure in this application to store and manipulate a collection of registered members in the system, and you must use *MovieCollection* as a data structure in this application to store and manipulate a collection of movies in the system.

5. Project Submission

- Each group should submit only one copy of your group work through the blackboard. Your group submission should be a single zip file named by *group-name*.zip and comprises of a complete C# Console Application project and a technical report in PDF.
- Your group submission must be submitted via the Blackboard. Email submissions are not accepted as any email having a C# project as an attachment may be blocked by the QUT

email server, and when this occurs neither the sender nor the receiver will be sent any notification.

- You may resubmit your assignment as many times as you wish before the deadline. If you submit your assignment multiple times, we will only mark the last submission before the deadline.

6. Assignment Marking

There are two components in the assignment marking. One is project marking, which weighs 75%; another is peer-review/evaluation, which weighs 25%. Your project submission will be marked by a tutor. If all the members in your group made roughly equal contribution to the project, all the members in your group will get the same marks from the project component.

Your performance in the group will be evaluated by all the other members in your group. This peer review/evaluation will be conducted online in the CAB301 Blackboard in Week 13. When the peer review/evaluation is open, we will send you a notification by email.

The full mark for this assessment is 40, including 30 marks from the project and 10 marks from the peer review/evaluation. The marking criteria and marks allocation for the project and the peer review/evaluation can be found in a separated document, namely Marking Criteria Sheet.