COMP10120 Practical Set 6: Sequential and Random-Access File Processing

Please read the questions carefully. Name each program based on your student number, the practical set number and question number. For this set (set6), question 1 should be named 1234567s6q1.c where your student number replaces 1234567. All questions that you are submitting can be zipped into a single file called 1234567s6.zip, where 1234567 is your student number and s6 refers to set 6. Please also include a readme.txt file which says which compiler you used to test your implementation. This zipped file can be submitted via Moodle for grading.

Part 1

1. Write a C Program which prompts the user for employee records and then writes each record to a file called *students.txt.* The user should be prompted to enter each student's first name, surname, student number, phone number, field of study, and GPA. Be careful to provide code to allow a user to indicate they have stopped inputting data and to check for invalid data. The result should be a text file with several student details.

2. Write a C Program which reads in the contents of the sequential file students.txt (created in question 1 above). It should read in each record and if the GPA is greater than 3.20, print that record to the screen in a neatly formatted way otherwise it should print a series of star symbols for that record.

3. You are a hipster and you have just opened a retro music record lending shop with a library of thousands of records. Customers can borrow records from you. Write a sequence of C Programs which will allow you to set up a system to record/catalogue the records that you have in a file. It should be possible to record details such as Title, Artist and year but also detail if the record has been borrowed and by which customer and when. One program should provide functionality to update these details. Use **Structs** and **Random Access Files** to achieve this.

Portfolio Ideas

1. Extend the programs written in Question 4 above to have the following functionality:
   - List all records that have been borrowed from the library.
   - List the ratio of records on loan to all records in the library
   - Search for records given an Author or a Title.
   - Copy all records and details to a sequential access file for printing in a nice format.

2. Using the **person struct** given below, write a C Program which prompts the user to complete the fields in the struct for 6 students. The program should then print out a list of the structs sorted numerically by gpa to rank the students.

```
struct person {
      unsigned int studentNum;
      char lastName[15];
      char firstName[10];
      char fiedOfStudy[50];
      int age;
      char sex; // ‘m’ or ‘f’
      double gpa;
};
```

NOTE: To make it efficient the program **must use** an array of pointers to structs. For this question, you may use *strcmp (string compare)* function from the C library (`#include<string.h>`) in your sorting function.