C Programming Lab
Assignment 3: Movies

Revision Date: September 10, 2014

Movie Database

How much digital data will the world create and store this year? Some place that number over 2 zettabyes! That's 21 zeros and is roughly the amount of storage of 57.5 billion 32GB Apple Ipads. Furthermore, the amount of data created and stored doubles every two years. How do we store and access this information? Usually with storage software known as a *database*. Databases are implemented everywhere from web servers to automobiles, video games, smartphones, etc. The goal of this project is to implement a simple C database and query system.

Tasks for your program:

- Display Movies
- Display longest movie
- Display shortest movie
- Display movies earlier than specified year
- Display movies later than specified year
- Display movies in the region of years specified
- Display movies of same actor
- Display movies released in same year
- Display movies of same length
- Add Movie
- Remove Movie
- Quit from program

Program behavior

Your program should use a menu driven interface that allows the user to interact with one table in a database Once your program is launched, it should read in the file of movie information and then display the following menu:

```
Welcome to the C Movie Database!
Main menu:
    1:  display all movies
    2:  display shortest movie
    3:  display longest movie
    4:  display older movies
    5:  display newer movies
```

```
    6:  display movies in the region specified
    7:  display all movies of same actor
    8:  display all movies released in same year
    9:  display all movies of same length
    10: add movie
    0:  quit the program
Enter option number:
```

The menu code (ideally a function call), should be placed in a loop. Only when the user enters a 0 should the loop be exited.

Option 1: Display all movies

Entering "1" will display the entire contents of the movie database. Based on the test *movies.dat* file, when you select the first option, your program should display something similar to:

```
The-Matrix Keanu-Reeves 90 1999
Pulp-Fiction John-Travolta 100 1994
Tombstone Kurt-Russell 100 1993
Blade-Runner Harrison-Ford 110 1982
Dark-Knight Bale 155 2008
Alien Sigourney-Weaver 95 1979
The-Shawshank-Redemption Tim-Robbins 115 1994
The-Wizard-of-Oz Judy-Garland 120 1939
A-Beautiful-Mind Russell-Crowe 180 2007
Titanic Leonardo 50 1998
Dark-Knight-Rises Bale 170 2012
```

Option 2: Display shortest movie

This option displays the shortest movie in the database. You will need to search your list of movies to find the shortest one. Note that your program will be tested on other databases.

Option 3: Display longest movie

Like Option 2, but displays the longest movie.

Option 4: Display older movies

When the user selects this option, the program queries the user for a year:

```
Enter option number: 4
Display movies older than what year? 1940
```

The program then displays movies that are older than the given year.

Option 5: Display newer movies

Like Option 4:

```
Enter option number: 5
Display movies newer than what year? 1998
```

The program then displays movies that are newer than the given year.

Option 6: Display movies in the region of years specified

Like Option 5:

```
Enter option number: 6
Display movies in between what years? 1998 2011
```

The program then displays movies in between these years (inclusive).

Option 7: Movies of same actor

Like option 6, your program will prompt for the actor

```
Enter option number: 7
Display movies of which actor? bale
```

and all movies of actor 'bale' will be displayed.

Option 8: Movies released in same year

Like option 8, your program will prompt for the year

```
Enter option number: 8
Display movies of what year? 2008
```

and all movies of that year will be displayed.

Option 9: Movies of same length

Like option 9, your program will prompt for the length

```
Enter option number: 9
Display movies of what length? 170
```

and all movies of that length will be displayed.

Option 10: Add a movie

Your program should prompt the user as follows:

```
Enter option number: 10
Adding movie...
    Title: New-Movie
    Main actors: Michael-Jordon
    Year: 2012
    Length in minutes: 145
Movie added.
```

In the example, the user has entered the text `"New-Movie"`, and so on. Selecting Option 1 again should display the new movie along with the others.

Option 11: Delete a record

Your program will prompt for the movie name

```
Enter option number: 11
Display movie name? Dark-Knight
```

and the related information of that movie should be removed.

Program organization

Create a *movies* directory off of your *clab* directory and move into *movies*.

Name your main c file *dbase.c*. This should contain only main function. All other functions should be in *util.c* file. You should have atleast 10 functions in *util.c* file.

Compliance Instructions

To make sure that you have implemented your program correctly, retrieve the files *movies.dat* from LMS or dropbox.

You should compile and link the files using following command.

```
gcc -c dbase.c
gcc -c util.c
gcc dbase.o util.o -o dbase
```

If your code fails with a runtime error while running this test, then you will receive a zero for this assignment.

Note that your answers do not have to be correct for your program to be graded, only that it not fail. Of course, correct answers will yield a much higher grade.

Implementation strategies

Start out by writing a program that can read and print out the database using *fscanf* and *printf* calls. Once you can successfully read and print the database, implement a menu system that handles the 0 and 1 options. With that working, implement each remaining option in turn, testing thoroughly before proceeding on to the next option.

Submission Instructions

Go to the *movies* directory containing your assignment. Do an *ls* command. You should see something like this:

```
dbase.c util.c dbase.o util.o dbase movies.dat
```

Extra files are OK. Submit your program like this:

```
submit clab mr movies <your-iiitb.org-email-address>
```

Due Date

October 15, 2014, 11:59:59 AM.