CS2308 - Foundations of Computer Science II
Program Assignment #2
Total Points: 70

Due 1 second before class begins, Mar. 10, 2020

In this project, you will write a program to simulate the ipod player. Your program will first initialize the ipod by reading in a number of songs from a file passed by the command line. Each song contains the title (string), artist (string) and the size (int) in MB. Your program should always prompt users the following window when executed:

Welcome to My Ipod!
Please choose your options:

1. Show the playlist
2. Remove a song
3. Cleanup my ipod
4. Shuffle the playlist
5. Exit

The Data Files

There is unknown number of songs in input_song.txt but your ipod only has 25MB of memory. When initializing the ipod, your program should keep reading in songs from the input file as long as the memory is not overflow.

Random Number Generation

Before generating a random number use argv[2] to seed srand() with an integer. Seeding the random number generator with a fixed number allows you to test the other parts of your code while fixing the order of numbers that will be generated.

Data Storage

You should store the songs in an array of ***Song*** structure as shown below. Since the size of the array cannot be determined until at runtime and you want to minimize the memory usage, you would have to use a dynamic array to store all the songs.

```
struct Song
{
        string title;
        string artist;
        float size;
};
```

Required Functions

Your program must have at least the following functions. You may create additional ones to further break the program down in to manageable units.

Main( )

Your main( ) function should be just a high level organizer. It should initialize your ipod by reading the input files, check if the file exists, set the random seed, prompt the welcome window to users unless the user choose to exit, call the right functions when users make choices. No other significant activity should take place in main( );

show_playlist ( )	shows the songs in the current playlist
remove_a_song( )	delete a song when the title is given. Report error if the title cannot be found. When a song is deleted, other songs may need to be shifted accordingly and the dynamic array size change as well.
cleanup( )	if the song list is not clean, delete all songs from the ipod and free the memory space. If the song list is already clean, your program should not free the dynamic memory again. Rather, the user should be told that the list is already clean (i.e. no songs in the list).
shuffle( )	shuffle the songs in the playlist with a random order. Should not show repeated order when calling this function multiple times.
exit ( )	if the song list is not clean, delete all songs from the ipod, free the memory space, then exit the program. If the song list is already cleaned up, just exit the program.

~~~~~~

Timeline:

Feb. 22:	main() is done. Stubs for other major functions done.
Feb. 25:	show_playlist() and remove_a_song() done.
Mar. 3:	cleanup() and shuffle() done.
Mar. 8:	All functions done. Start testing your programs.
Mar. 10:	Due Date. It should be fully tested, commented and follow the coding standards.

Comments and Suggestions:

DO NOT DELAY. Start writing the program from Day 1. If you wait until the night before the due date, you will have a miserable night and it is less likely you could complete the project.

Do not try and write the entire program all at one time. Work on the program in small sections, as the timeline indicates.

Notes:

- **Your code will be graded in Linux system, make sure to test your code on the Linux server.**
- Your program must correctly compile and is executable.
- Be sure your code file is readable and neat. Do not allow lines to extend past 80 characters, use appropriate white space and make sure to use a consistent indentation scheme shown in coding standard.
~~~~~~

Program Submission

Please submit only your source file (firstname_lastname_prog2.cpp) to TRACS. DO NOT submit the entire project! You will get zero if you fail to submit your cpp file to the web server before the deadline.

You also need to submit a paper copy of your source code in class at the deadline day as well. If you fail to submit the paper copy in time or your paper copy does not match your electronic copy, you will lose 20% of your total grade every day until the instructor received the correct version of your paper copy. Your program will be graded as follows, please make sure to check each item before you submit.

CS2308 Program #2 Name:_____

Program and Run Time Output: _____ (60 Points)

_____ (8) Welcome Menu

_____ (3) Welcome menu is shown correctly throughout the program

_____ (5) Functions are called correctly when user chooses an option

_____ (15) Ipod initialization

_____ (4) Open and check files, display message and exit upon error

_____ (2) Check the memory limitation while reading in songs

_____ (2) Read correct number of songs from the input file

_____ (5) Generate the dynamical array correctly

_____ (2) Read songs into an dynamic array of structure

_____ (5) Show the playlist correctly

_____ (10) Remove a song

_____ (8) Remove a song correctly when a title is found

_____ (2) Display error messages when a title is not found

_____ (5) Clean up the ipod correctly and free the memory

_____ (12) Shuffle the songs in the playlist

_____ (7) Songs are shuffled in random order

_____ (1) srand set by command line seed in main

_____ (4) dynamic array (via a pointer) is passed to the shuffle function correctly

_____ (2) correct function prototype

_____ (2) main ( ) correctly controls the entire program and display meaningful messages when errors occur

_____ (1) Program exit correctly

Coding Standards: _____ (10 Points)

_____ (1) Print the grading rubric page and staple it as page one

_____ (3) Documentation (program and function headers)

_____ (2) Comments

_____ (2) Meaningful Variable and Function Names

_____ (2) Indentation Scheme / Use of { }

Total: _____ (70 Points)

Executable Version: _____ (%)

If your code cannot be compiled or executed, your final score will only be 50% of your accumulated score of each item above.

Final Score: _____ (70 Points)