Playlist

Introduction

This assignment requires you to develop a program where a user can create a playlist of individual songs. You will implement a text based menu, this will allow users to edit the playlist. To gain full marks in this assignment you will need to implement the playlist using a linked list; an array-based implementation means that marks gained will scaled downwards by 30% so the overall grade will be out of a maximum of 70%.

Requirements

Each Song in the playlist should have the following attributes:

- the name of the song
- the name of the artist
- the duration of the song (in seconds)

The Playlist will contain Songs in an order imposed by the user. This can be implemented using a singly linked list for full marks or an array for a lower grade. Each position in the playlist should have the following attributes:

- the name of the playlist
- a pointer to a song
- a pointer to the next position in the playlist (if using a linked list implementation)

The text based interface allows the user to interact with the playlist. The following options should make up your text based interface:

1. Create a new Playlist - the user should have the option to give the playlist a name
2. Add a Song to the end of the Playlist - the user should be prompted to provide the details of the new song
3. Play the Playlist - print out each song in the playlist in order
4. Search for a Song based on the title or the artist's name - prompt the user to input a string, if this string matches, or is a sub-string of, either the title or artist of a song then return that song (these criteria could return multiple songs)
5. Delete a Song from the Playlist - the user should be prompted to select the the index of the song to be removed
6. Swap the position of two Songs - the user should be prompted to select the indexes of two songs which will swap positions in the list
7. Play a random Song from the list - use random function in C to randomly print out one of the songs

8. Shuffle the order of the Playlist - reorder the songs of the playlist with some degree of randomness and print out the new order (see note below on shuffling)
- shuffle in place
    - shuffle with a copy
9. Calculate the duration of the Playlist - print the total cumulative time of all songs in the playlist
10. Save the complete Playlist to a file
11. Load a complete Playlist from a file