

The project is to create a music library program that will enable users to create a music library that will accommodate both popular and classical music. The items of data that you will need to keep are different for the two kinds of music. The following table lists the various data items that you will need to keep for each type of music.

Popular	Classical
Song Title	Piece Title
Artist (Performer)	Movement Title
Album Title	Movement Number
Genre	Composer
File Name	Conductor/Performer
	Album Title
	File Name

The library will consist of a linked-list of objects. The elements of the linked list will be objects defined as a polymorphic type. The list will be doubly-linked to permit the deletion of objects.

The program will accept a number of different commands from the user. These are:

1. open <file name>
This will read a new file into the program and create a new linked list or Add to the existing linked list.
2. close
This will destroy the existing linked list and recover the storage it occupies.
3. print <option>
where the options are “title”, “artist”, or “album”
this command will print the entire list (on cout) in order by Song/Piece Title, Artist/Composer, or Album.
4. add <option>
where option is “classical” or “popular”
The program will issue a number of prompts for the required information
5. display
The current element of the linked list will be displayed. This, by default, is the First item of the list, but it can be changed using the first, last, next, or previous Commands
6. first, last, next, previous
these commands change the current element of the linked list. No error messages will be issued by these commands. The action of each should be self-explanatory. If you are at the end of the list, “last” and “next” do nothing. Similarly if you are At the beginning of the list, “first” and “previous” do nothing.
7. save <file name>
This will write out the current linked list to a file. If the file name is omitted, the Last file mentioned in an “open” command will be used. If no such command has Been used, then the name “default” will be used.

8. delete

this command will delete the current list element and set the current list element to the next element, or the last element in the list if the last element is deleted.

The format of the files read and written by this program will be as follows.

The first line of any entry will be the word “Popular” or “Classical” followed by a newline. Next will come the data for the fields, one line per item. The order of the items will be as given in the table above. The following is an example of what a file would look like.

```
Popular
True Colors
Cindy Lauper
True Colors
Rock
C:\My Music\True Colors.mp3
Popular
Walk of Life
Dire Straits
Brothers in Arms
Rock
C:\OtherMusic\wl.wav
Classical
Suite 1 for Cello
Allemande
2
J. S. Bach
Yo Yo Ma
Six Suites for Cello
D:\LotsaMusic\Ma.mp3
Classical
Symphonie Fantastique Op. 1
Marche au Supplice
4
Hector Berlioz
Leonard Bernstein
Symphonie Fantastique
D:\OtherFiles\Marche.wav
```

Due dates:

Flow charts, design of object structure: July 20th

Final project, code plus user's manual: Aug 3rd