
HW 7: Doubly Linked Lists

Write a program that declares `structs list`, `node`, and `data`, as described on the 12_linkedlists slide set, to store the data of a football player (name, position, scores, catches, passing yards, receiving yards, rushing yards). Use a doubly linked list data structure (with a header) based strategy as discussed in class. The program should accommodate a variable amount of players (records). The database should be kept in a text file, `hw07db.txt`, which is used for both input and output. Create a menu based interface with options for adding/removing a record to/from the database, displaying one or all records on the screen, and saving the list to the text file `hw07db.txt`.

Where the text file:

- has seven fields per record (one per line)
- has an empty line between records (last line is empty)
- resides in the same folder with the other relevant project files
- as mentioned above, is called `hw07db.txt`

Example menu screen:

Select from the following options:

1. Add a record
2. Delete a record
3. Display a record
4. Display all records
5. Save and exit

Enter your choice:

Do not use STL container variables (`vector`, `deque`, etc). Use value returning and/or non-returning functions which are passed value and/or reference parameters. Use whatever function return types and/or parameters you prefer, but keep in mind that a better design will have higher grade. "Better design" means: minimizing (1) workload on client code, (2) recompilation upon individual changes, (3) memory [spatial] requirements.

You have to:

1. Adhere to the usual nomenclature (hw07...)
2. Have a **Makefile**
3. Use a header file (**hw07.h**) for assignment and your personal information as usual
4. Use a separate header file, **hw07f.h**, for declarations of functions (listed below)
5. **#include** both header files in the file that contains your **main()** (**hw07.cpp**)
6. Use functions for handling each menu task
7. Put the functions in separate files:
 - **menu()** // in **menu.cpp**
 - **popList()** // in **poplist.cpp**
 - **addRec()** // in **addrec.cpp**
 - **delRec()** // in **delrec.cpp**
 - **showRec()** // in **showrec.cpp**
 - **showRecs()** // in **showrecs.cpp**
 - **saveExit()** // in **exit.cpp**
8. use **valgrind** to make sure there are no memory leaks or errors
9. include the **valgrind** run in your **hw07.scr** file

Organize it as usual; use the CLI **script** command to capture an interaction with the program adding at least five players. Exit the program. Start it again. Display all, delete the third, display all, delete the first, display all, and display one by letting the user specify the respective name (which implies a search). Exit the program.

Package a **tar** file that includes **Makefile**, **hw07.h**, **hw07f.h**, **hw07.cpp**, **hw07db.txt**, **menu.cpp**, **poplist.cpp**, **addrec.cpp**, **delrec.cpp**, **showrec.cpp**, **showrecs.cpp**, **exit.cpp**, and **hw07.scr**. Submit it to canvas by the due date on top of this page.