**Programming Assignment 4**

Structures, arrays of structures, functions, header files

Program description:

1. **Complete** the program provided to read football statistics into a structure array, offers a menu to update statistics, print data, and update a file.
2. The program declares a struct to store the data of a football player and has the following data members. The structure in the header file provided:

Name

Position

Number of Touchdowns

Number of Catches

Number of Passing Yards

Number of Receiving Yards

Number of Rushing Yards

1. An array of the structure with 30 components is declared to store the data of football players.
2. A function is provided to open the input file.
3. A function skeleton is provided to read the data in the file “footballdata.txt” into the structure array.
4. An Integer function is provided to search the structure array to find the index of a specific player.
5. A function has been provided to using the index returned from the search, print all information for a player using the index returned from the search.
6. A function has been provided to print all data for all players.
7. A menu is provided to allow the user to update the data for a player, print player data or print all the data. The menu may be in main. The program **loops** until the user selects 99.
8. Before the program terminates, a selection to give the user the option to save data in a file, name of your choice, is provided. The user enters the file name. A function to open the output file and check the status has been provided. If the file does not open, a message is displayed to the user that the data will not be written to the file and skip the output to the file. **Do NOT** overwrite the input file.
9. All files are closed prior to program end.
10. A menu is provided to select one of the following options:
11. Print a player's data
12. Print all the data
13. Update a player's touch downs
14. Update a player's number of catches
15. Update a player's passing yards
16. Update a player's receiving yards
17. Update a player's rushing yards

99. Quit

1. **Your tasks:**
   1. **Individually (not the group).** In the function **getData**, read the data in the attached file, footballData.txt, into the provided structure array until end of file. While reading, count the number of items read and return from the function. This will be used as the limiting number of the array. **Call the function from main**.
   2. Retrieve functions from GitHub to complete the program.
   3. **Individually (not the group).** Include the function prototypes in the header file. Include the header file in the program. DO NOT CODE THE FUNCTIONS IN THE HEADER FILE. Functions MUST be coded in the .cpp file AFTER main.
   4. **Individually (not the group).** Submit a running program, free of compiler errors. Submit in the Programming Assignment 4 Link.
   5. **Individually (not the group).** Download and complete the CSC-Practical Exposure to Career II Assignment Overview Reflective Response document. Submit the complete document in the same link, not in the Programming Assignment 4 link.
   6. **For all groups**: To perform the processes in the menu, create the following functions. **All** functions should be called for each of the menu options selected.
2. For each of the update functions, request the amount to update from the user.
3. Also ensure that the index passed into the functions is valid, i.e., contains a valid index within the bounds of the array using the number of components in the array as a comparison (error checking)
4. Each group’s function must have required documentation (pre and post conditions).
5. Using Zoom, collaborate with group members on how to design your group’s function. You can also use the collaboration/discussion functionality in GitHub to have discussions off-line with your group. You will receive emails from GitHub with discussion contents. There is also a discussion forum in canvas you may use for your group.
6. Upload function to GitHub.
7. **All functions must be completed one (1) week prior to program due date.**
   1. **Group 1**: Using the index returned from the search, update the number of touchdowns for the player. Pass in the index, the structure array, the value to update touchdowns and the number of components in the array. Update means add to the existing value.
8. **Group 2**: Using the index returned from the search, update the number of catches for the player. Pass in the index, the structure array, the value to update catches and the number of components in the array. Update means add to the existing value.
9. **Group 3**: Using the index returned from the search, update the number of passing yards for the player. Pass in the index, the structure array, the value to update passing yards and the number of components in the array. Update means add to the existing value.
10. **Group 4**: Using the index returned from the search, update the number of receiving yards for the player. Pass in the index, the structure array, the value to update receiving yards and the number of components in the array. Update means add to the existing value.
11. **Group 5**: Using the index returned from the search, update the number of rushing yards for the player. Pass in the index, the structure array, the value to update rushing yards and the number of components in the array. Update means add to the existing value.

**Test and evaluate calculations for accuracy. Points will be taken for inaccurate calculations, improper formatting, directions not followed.**

Turn in file code \*.cpp; README file; header file and the input files. You DO NOT need to submit the executable (\*.exe). You DO NOT need to submit the output file. It will be created when I test the programs. You MAY zip all the files and submit if you choose. You MUST include all the files indicated or points will be deducted.

README must contain instructions for location of input/output files

**Points WILL BE taken if MINIMUM requirements and submissions not included.**

**1 point extra credit will be awarded if assignment is zipped for submission**

Each solution is to be uniquely your own; minimal student collaboration allowed.

See Canvas and the Syllabus for due date