CIS 278 (CS1) Programming Methods: C++

Assignment 8: Structs

Skip to Main Content

Learning Objectives

After the successful completion of this learning unit, you will be able to:

- Solve problems using structs in accordance with good programming practice.

Assignment 8.1 [45 points]

Rewrite your most recent high scores program so that each name/score pair is stored in a struct named highscore. Except as noted below, this new program will continue to meet all of the requirements of your most recent high scores program. Your new program should meet the following requirements:

1. The highscore struct should have two fields:
    - an int named score
    - and a char array named name. The char array should have 24 elements, making the maximum length of the name 23. (If you prefer to use a char pointer and a dynamically allocated array, that is fine as well. However, this may result in a number of complications, so be prepared for the challenge.)
2. The data should be stored in a single array, a dynamically allocated array of highscore structs.
3. Your program should use three functions that accept the array of highscore structs:

```
void initializeData(highscore scores[], int size)
void sortData(highscore scores[], int size)
void displayData(const highscore scores[], int size)
```

4. You may use any sort algorithm, but I would recommend using the selection sort from **lesson 9.6**. Don't use C++'s sort() function, but you can use the swap() function.
5. Note that when you swap your array elements, you can swap the entire struct. You don't need to swap the name and the score separately.
6. You may assume that the user enters names that are 23 characters or less. Getting this to work correctly if the user enters names that are too long -- that is, making it so that you put the first 23 characters in the name variable and ignore the remaining characters on the line -- is complicated. You can do this as an extra challenge if you want, but it's not required.

Name your source code file according to the assignment number (a1_1.cpp, a4_2.cpp, etc.). Execute the program and copy/paste the output that is produced by your program into the bottom of the source code file, making it into a comment. Use the Assignment Submission link to submit the source file. When you submit your assignment there will be a text field in which you can add a note to me (called a "comment", but don't confuse it with a C++ comment). In this "comments" section of the submission page let me know whether the program works as required.

Keep in mind that if your code does not compile you will receive a 0.