

Objectives: You will gain experience using C++:

- defining functions
- arrays
- parameter passing of arrays

Download the following file to your desktop: <http://www.cs.uni.edu/~fienu/cs051f09/labs/lab8.zip>

Extract this file by right-clicking on lab8.zip icon and selecting Extract All.

Part A: Yesterday in lecture, we developed a function `locateLargestScore` that took as parameters an array of scores and the count of scores, and returned the index location of the largest score in the array. The lab8.zip file you downloaded and extracted contains a `ProcessScores` folder with a Visual Studio C++ project file: `ProcessScores.sln` inside. Double-click on it to start this project in Visual Studio.

a) Your first task is to create a **new function** `lowestScore` that determines the value of the lowest score (**not the location** of the lowest score). Add a call to this new function `LowestScore` in the main program.

b) Assume that your instructor occasionally misgrades a question on an assignment or test, and decides to increment (or decrement) everybody's score by a fixed number of points. Create a **new function** `modifyAllScores` that take as one of its parameters an amount to adjust all the scores by (this can be a positive or negative value). Add a call to this new function `modifyAllScores` in the main program.

After you have both functions for Part A complete and working correctly, raise your hand and we'll check your work.

Part B: The main program of the `processScores` application would be improved by:

- a) moving the first while-loop to a function `getScores`, and
 - b) moving the second while-loop to a function `possiblySaveScores`
- Add calls to both of these new function in the main program.

After you have both functions for Part B complete and working correctly, raise your hand and we'll check your work.

Nothing needs to be turned in for this lab. Make sure that you log off the computer before you leave.

EXTRA CREDIT:

If you have time to kill and/or want some extra credit, modify the main program to include a menu-driven driver program. I'd call the `getScores` function once before going into the menu-driven driver loop, after that all of the score related functions should be options in the menu.