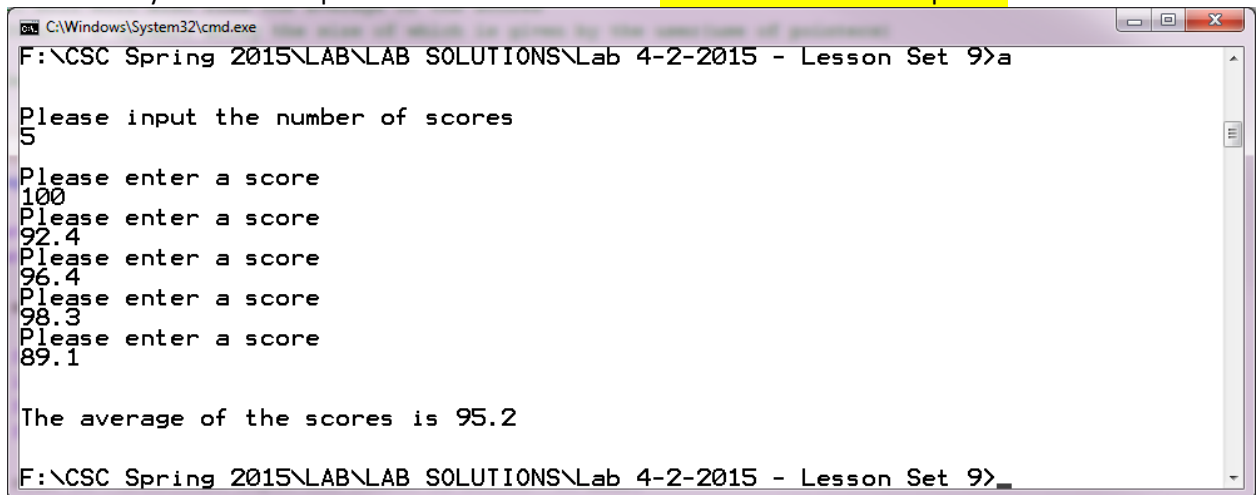


LAB DIRECTIONS for LAB 4/02/2015

Based on Lesson Set 9

1. Create a folder named **Lesson Set 9**. Put all files you create for this lab in this folder.
2. There will be no source files provided for you for this lab. You will be creating all files yourself.
3. Write a program named `score.cpp` that will read scores into an array. The size of the array should be input by the user (dynamic array). The program will find and print out the average of the scores. Please read the specifications for this program below:
 - a. Use a programmer defined function named `makeArray`. You should send the number of scores to this function and return a pointer to a dynamically created array.
 - b. Have the user enter the scores in the main function. Make sure to use pointer notation.
 - c. Call a function named `findAverage` to find and return the average of the scores in the array. You should send the array and the number of scores to this function. Make sure to use pointer notation in the `findAverage` function.
 - d. Print the average of the scores with only one place after the decimal to the screen in the main function.
 - e. Make sure to release the memory that was used for the dynamically created array.
 - f. Make a screen capture of this program running using the same user input as in my screen capture below. Put your screen capture in a document named **Lesson Set 9 Screen Captures**.



```
C:\Windows\System32\cmd.exe
F:\CSC Spring 2015\LAB\LAB SOLUTIONS\Lab 4-2-2015 - Lesson Set 9>a

Please input the number of scores
5

Please enter a score
100
Please enter a score
92.4
Please enter a score
96.4
Please enter a score
98.3
Please enter a score
89.1

The average of the scores is 95.2

F:\CSC Spring 2015\LAB\LAB SOLUTIONS\Lab 4-2-2015 - Lesson Set 9>_
```

4. Use the program you created above to complete this next assignment.
 - a. You are going to separate `score.cpp` into three files:
 - `scoreB.h` – contains `#include` files, using namespace & function prototypes
 - `scoreB.cpp` – contains main function only
 - `functions.cpp` – contains all programmer defined functions

- b. View a skeleton version of scoreB.h below:

```

1  /*
2      FULL COMMENT BLOCK HERE
3  */
4  #ifndef SCOREB_H
5      #define SCOREB_H
6
7      //files to be included
8
9      using namespace std;
10
11     //function prototypes
12
13 #endif
14

```

- c. Your source files scoreB.cpp and functions.cpp should both include ONLY ONE #include at the top under your comment block as you see below:

```
#include "scoreB.h"
```

- d. Put a full comment block at the top of ALL THREE files. The only thing that is different in the comment block of each file is the title of the file.
- e. Compile & run your program. Here is the command for compiling these three files together:
`g++ -I ./ functions.cpp scoreB.cpp -o scoreB.exe`
- f. Make a screen capture of this program running using the same user input as in your first program. Your output should be exactly the same. Make sure I can see how you COMPILED & RAN the program so that I can see that you compiled all three files together. Put your screen capture in a document named **Lesson Set 9 Screen Captures**.

What to Turn In: (by Wednesday, April 8, 2015)

- score.cpp
- scoreB.cpp
- scoreB.h
- functions.cpp
- Lesson Set 9 Screen Captures

How you will be graded

score.cpp	30 points	FOLLOWS SPECIFICATIONS / AVERAGE is CORRECTLY CALCULATED / FUNCTIONS ARE CORRECT / CODE IS IN PONTERR NOTATION
scoreB.h	20 points	FOLLOWS SPECIFICATIONS – only includes comment block, includes all header files, includes function prototypes, does not include any function definitions
scoreB.cpp	20 points	FOLLOWS SPECIFICATIONS – only includes comment block, #includes header file, and main function
functions.cpp	20 points	FOLLOWS SPECIFICATIONS – only includes comment block, #includes header file, and programmer defined functions
Lesson Set 9 Screen Captures	10 points	Screen capture for two programs was included (5 points each)