Pair Programming 5 Activities

* **Always use the pair programming tests to ensure your program works properly.**
* **Take a screen shot of each execution in the tests.**
* **Download the source code file for inclusion in the turn in document.**
* **Turn in pair programming activities using the pair programming turn in document.**
* **It is each individual’s responsibility to test all programs and create your own turnin document. Pair programming is graded individually, so be sure to share the work you and your partner did together as you go.**
* **You can only earn 50% of pair programming points if you do not work with your partner unless there are extenuating circumstances.**
* **Do not share work with your partner that you did not do together.**
* **Review the Test document for this lab to see the user interaction and displayed text that is expected by the code that you write. Please limit the creativity you use on this assignment.**

5a: (2 points) In a C++ file called pp5a.cpp, write a function called *myswap* that takes 2 integer parameters and swaps the values of the actual parameters. Its post-condition is that the actual parameter values have been swapped which means **the formal parameters must be reference variables.** If the formal parameters are not reference variable, the function will not behave properly. Write a driver (a main function that calls myswap) to test the myswap function.

5b: (5 points) In a C++ file called pp5b.cpp write a single function called *figureIt* that determines the following attributes of a floating-point number:

1. Is it evenly divisible by 2? Result should be a Boolean value
2. What is its sign? Result should be the character + or –
3. What is its integer part rounded up or down as appropriate? Result should be an integer. Do **not** use the built-in *round* function.

Since the function has to “return” these three values, it can’t use a return statement to do this. Return statements can only return one value. Therefore, the function will have to “return” these three values using reference variables. The function has a total of four parameters and a post-condition that three of the actual parameters are changed by the function. The function *figureIt* does not have any input or output in it. Write a driver (a main function) that loops to get a real number from the user then calls your function. The loop should stop when the user enters 0. The main function should print the results acquired by the *figureIt* function.

5c. (3 points) In a file called pp5c.cpp write a function called *getData* that has two parameters, a name and an age. The function gets a user’s first name and age from standard input. The function “returns” the name and age via its parameters. The function’s post-condition is that the actual parameters are altered. Write a driver to call this function to test it. The driver should print the name and age after the call to getData. To store a name, you can use the *string* data type.