**Scoring Rubric for Project 3 : BubbleSort**

*Due 10/03/2019 @ 3:30 pm*

|  |
| --- |
| Student Name: Sarah Ruth Nicholls |

|  |  |  |
| --- | --- | --- |
|  | **Score** | **Maximum** |
| **Execution (50 pts):** | | |
| Program compiles without errors (warnings are okay) | 50 | **50** |
| **Implementation (40 pts):** | | |
| Uses function declarations as provided | 5 | **5** |
| Main function includes at least one unit test for Swap (can use assert or printed output) | 5 | **5** |
| BubbleSort works for input size of 42 and 47 (all or nothing) | 5 | **5** |
| Use a dynamically allocated array for BubbleSort | 5 | **5** |
| Free the allocated array at the end of Main function | 2.5 | **5** |
| Complete the BubbleSort unit test | 5 | **5** |
| Use command line arguments to read the array size and the seed | 5 | **5** |
| Measure the execution times of MergeSort and BubbleSort and plot them on a graph | 5 | **5** |
| **Style (10 pts):** | | |
| The driver and functions are easy to follow based on the use of comments | 0 | **6** |
| Easily identifiable variable names | 4 | **4** |
| **Total (100 pts):** | 91.5 | **100** |

Notes:

You need to add array = nullptr; to finish deallocating the array. This ensures array is not left as a dangling pointer.

Not sure why your unit test for swap was commented out, but I counted it. For future reference, this unit test should go before bubblesort because you want to make sure it works correctly before using it in a function.

You didn’t add any of your own comments.