Self-Evaluation Report for Programming Assignments
Section I: Basic Information

1. Programming assignment # __3_____

2. Name of the author: _____Mark Nguyen_____

3. Name of the peer reviewer if any: _____

4. Due date of the assignment: _____12/8_____ 5. Date when the assignment was finished: _____NA_____

6. Number of hours spent in programming: _____4_____

Section II: Integrity Review

Integrity rules for regular programming assignments

● Peer discussion: Peer discussion of code shown on a screen or board is acceptable for explanation of ideas and for debugging purpose. Such discussion may help to cultivate an open learning environment in the class, but you should carefully read the guidelines below to avoid any dishonest behavior and never step over the guidelines explicitly described in the following.

● **Never use any code (i.e. C++ statements, segments of a program or an entire program) written by others (except for examples in our textbooks or reading): Any copy-and-paste of code from other people's programs or from websites is viewed as cheating and you will get 0 points for the assignment.**

● Never circulate your code to others: You should never pass around your code (electronically or on paper) to others except for the TA and the instructor. Violating this rule is viewed as cheating in the class and the provider will receive 0 points for the assignment.

● Never provide false or exaggerated results of test cases: You need to report results of test cases in the self-evaluation report together with all your source code files for each assignment. Providing false or exaggerated results of test cases in the report is viewed as cheating and you will receive 0 points for the assignment.

● Demonstrate the credibility of your authorship of the work: When you submit your code as your own work for points, you should make sure that you are able to explain your code and reconstruct your code from scratch without any outside help when requested. If you are not able to do that on your own when requested, you will get 0 points for the assignment and there will be an investigation.

● Consequence of cheating in the class: Cheatings end in 0 points for the assignments followed by discipline actions described in the student handbook.

1. Have you ever received any code written by others?_____no_____ 2. Have you ever passed any code you wrote to others? __no___ 3. Have you ever used any code written by others?_____no_____

Section III: Test cases and peer review

Note: To get all the points, you should have a peer reviewer watch the behavior of your program before you submit the work. You should prepare your own test cases and have your reviewer see the results when you run your program over the test cases. Optionally, you may also have the reviewer run your program through the reviewer's own test cases to see whether your program works correctly.

1. Compile and run your code using Visual C++ 2013 as the testing environment and describe the test cases used and the results you and the reviewer have observed:

Read value;

2. Description of bugs or other problems discovered by you or the peer reviewer if any:

3. Have you implemented everything required by the programming assignment? If not, describe what are missing.

I did not finish the DISPLAY or ASSIGNMENT functions, only the read

Section IV Self-evaluation: Points you think you deserve _____5_____

● Deduct one point if you submit the work after the due date but before it's closed.

● Grading scale:

0. Nothing done or missing the self-evaluation report or missing the integrity review in the report

1. Source code is completed but the code fails to compile successfully

2. Source code can compile and do something required, but has serious bugs or miss a couple of key features.

3. Source code can compile and do most of the features required, but has many minor bugs or miss a key required feature.

4. Source code can compile and do all the features required, nearly fully functional, only a couple of minor bugs.

5. Source code can compile and do all the features required, fully functional, no bugs.

6. In addition to the points received according to the rubrics above, get one more point if

a. the self-evaluation report contains sufficient descriptions of test cases used (0.25 point), and

b. the self-evaluation report indicates the results of the test cases were verified by a peer reviewer (0.25 point), and

c. the source code is well indented and commented to make it visually very readable (0.5 point).