**Code Review 6433**

**Overview**

In this testing plan, we are to perform black box testing with the 18 provided test cases using 6433’s code.

**Test Cases**

**Test Case 1 (Unsorted Table)**

- *Testing a table sorter to sort an Unsorted Table.*

- ***Pass****: Yes*

**Test Case 2 (Sorted Table)**

**-**  *Testing a table sorter to check if the sorted table is returning true.*

-  ***Pass****: Yes*

**Test Case 3 (Table with max value of integer)**

**-**  *Testing a table sorter to sort an Unsorted Table with the max value of an integer.*

**- *Pass:*** *Yes*

**Test Case 4 (Table with max + 1 value of integer)**

**-** *Testing a table sorter to sort an Unsorted Table with the max value + 1 an of integer.*

**- *Pass:*** *Yes*

**Test Case 5 (Table with all zeros)**

**-**  *Testing a table sorter to check if a table of all zeros returns true.*

**- *Pass:*** *Yes*

**Test Case 6 (Unsorted table of zeros and ones)**

**-** *Testing a table sorter to sort an Unsorted Table with 1s and 0s.*

**- *Pass:*** *Yes*

**Test Case 7 (Table of all ones)**

**-** *Testing a table sorter to sort and check if a table containing all 1’s returns true.*

**- *Pass:*** *Yes*

**Test Case 8 (Table of all tens)**

**- *Testing a table sorter to check if a table of all 10s returns true.***

**- *Pass:*** *Yes*

**Test Case 9 (Unsorted table with 1, 2, 3, and 4 digits of numbers)**

**-** *Testing a table sorter to sort an unsorted table container 1, 2, 3, and 4-digit numbers to test if the program can handle digits bigger than 1.*

**- *Pass:*** *Yes*

**Test Case 10 (Unsorted table with a -1)**

**-** *Testing a table sorter to sort an unsorted table with negative values.*

**- *Pass:*** *Yes*

**Test Case 11 (Table with the character ‘a’)**

**-** *Testing a table sorter to see if it sorts letters.*

**- *Pass:*** *Yes*

**Test Case 12 (Table with characters like ‘@’, ‘/’ and ‘[‘)**

**-** *Testing a table sorter to sort special characters.*

**- *Pass:*** *Yes*

**Test Case 13 (Table with the minimum value of integer)**

**-** *Testing a table sorter to sort the minimum value of an integer.*

**- *Pass:*** *Yes*

**Test Case 14 (Table with the minimum – 1 value of integer)**

**-** *Testing a table sorter to check if it can handle the minimum value +1 of an integer.*

**- *Pass:*** *Yes*

**Test Case 15 (Table with a non-square input)**

**-** *Testing a table sorter to check handles inputs that aren’t squares.*

**- *Pass:*** *Yes*

**Test Case 16 (Table with decimal values)**

**-** *Testing a table sorter to check if the table sorter handles doubles.*

**- *Pass:*** *Yes*

**Test Case 17 (Table with fractions)**

**-** *Testing a table sorter to check if it handles fraction doubles.*

**- *Pass:*** *Yes*

**Test Case 18 (Table with a string)**

**-** *Testing a table sorter to check if it handles strings.*

**- *Pass:*** *Yes*

**Test Cases and Code Conclusions**

After performing the tests, we can conclude that 6433’s code worked and following 4680’s testing plan lead to a perfect success score on all test cases. While 6433’s code did work, it was a bit difficult to follow code structure. This had no impact on the test and how they performed but, it was a bit difficult to set up and test the code. Comments were provided and clearly and effectively explained what each method did and even provided a way to test the code using a text file. Overall, 6433’s code was successful in running and executing the tests with no problems.