# System Test Plan for SortedDoubleArray

The text in bold in the **Description** column represents the test inputs to the program. You can stop program execution by pressing Ctrl+C. You will need to fill in your Actual Results and write the two additional tests (highlighted in yellow).

| Test ID | Description | Expected Results | Actual Results |
|---|---|---|---|
| Test invalid number of values (double and String) and unsorted array.<br><br>Author: Dr. Schmidt | *Preconditions:* SortedDoubleArray program started<br><br>Enter number of values to store: **1.0**<br>Enter number (integer) of values to store: **two**<br>Enter number (integer) of values to store: **3**<br>***Initial:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: [0.0, 0.0, 0.0]<br>Enter double: **5**<br>Enter double: **7**<br>Enter double: **2.0** | ***After all values added:***<br>Array that adds elements in sorted order: [2.0, 5.0, 7.0]<br>Array that adds elements in the input order, then sorts: [5.0, 7.0, 2.0]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.0, 5.0, 7.0]<br>Array that adds elements in the input order, then sorts: [2.0, 5.0, 7.0] | ***After all values added:***<br>Array that adds elements in sorted order: [2.0, 5.0, 7.0]<br>Array that adds elements in the input order, then sorts: [5.0, 7.0, 2.0]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.0, 5.0, 7.0]<br>Array that adds elements in the input order, then sorts: [2.0, 5.0, 7.0] |
| Test negative number of values and zero values<br><br>Author: Dr. Schmidt | *Preconditions:* SortedDoubleArray program started<br><br>Enter number of values to store: **-9**<br>Enter number of values to store: **0** | ***Initial:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: []<br>***After all values added:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: []<br>***After sorting second array:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: [] | ***Initial:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: []<br>***After all values added:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: []<br>***After sorting second array:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: [] |

Benjamin Uy and Alex Calisto

| Test ID | Description | Expected Results | Actual Results |
|---|---|---|---|
| Test valid number of values and sorted array<br><br>Author: Benjamin Uy | *Preconditions:* SortedDoubleArray program started<br><br>Enter number (integer) of values to store: **4**<br>***Initial:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: [0.0, 0.0, 0.0]<br>Enter double: **2.2**<br>Enter double: **4.0**<br>Enter double: **6.0**<br>Enter double: **8.8** | ***After all values added:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8] | ***After all values added:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8] |
| Test valid number of values and reverse sorted order<br><br>Author: Benjamin Uy | *Preconditions:* SortedDoubleArray program started<br><br>Enter number (integer) of values to store: **4**<br>***Initial:***<br>Array that adds elements in sorted order: []<br>Array that adds elements in the input order, then sorts: [0.0, 0.0, 0.0]<br>Enter double: **8.8**<br>Enter double: **6.0**<br>Enter double: **4.0**<br>Enter double: **2.2** | ***After all values added:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [8.8, 6.0, 4.0, 2.2]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8] | ***After all values added:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [8.8, 6.0, 4.0, 2.2]<br>***After sorting second array:***<br>Array that adds elements in sorted order: [2.2, 4.0, 6.0, 8.8]<br>Array that adds elements in the input order, then sorts: [2.2, 4.0, 6.0, 8.8] |

### Hint: Possible Test Cases

It may be helpful to consider user values that are input in sorted order or reverse sorted order.