**Code: 4478**

**Test plan: 0538**

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 1 | | Current Status: failed | |
| Test tittle: Checking if an empty table is sorted | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a table of size 0 and no values within. | Initial condition | Creation of an empty Table. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use assertTrue to test that the isSorted method returned true. | Force isSorted method to check an empty table. | Test passes. |

Given a table of size 0 with no values within, Test No: 1 returns “Table is empty!!!!” instead of returning true. The test failed since it did not return true.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 2 | | Current Status: Passed | |
| Test tittle: Sorted table of size one | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a table of size 1 and 1 value within. | Initial condition | Creation of a table of size 1. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use assertTrue to test that the isSorted method returned true. | Force isSorted method to check a table of size 1. | Test passes. |

Given a table of size 1 with the value ‘5’, Test No: 2 returns true. The test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 3 | | Current Status: Passed | |
| Test tittle: Sorted table of size 5 | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a sorted table of size 5 and 25 values within. | Initial condition | Creation of a sorted table of size 5. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use assertTrue to test that the isSorted method returned true. | check a sorted table of size 5 using isSorted. | Test passes. |

Given a sorted table with the following values: {1, 2, 3, 4, 5, 6, 7, 8, 9. 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25}, Test No: 3 returns true for a sorted table of size 5. The test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 4 | | Current Status: Passed | |
| Test tittle: Unsorted table of size 5 | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize an unsorted table of size 5 and 25 values within. | Initial condition | Creation of an unsorted table of size 5. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use assertFalse to test that the isSorted method returns false. | check a sorted table of size 5 using isSorted. | Test passes. |

Given the unsorted table with the following values: {5, 2, 7, 4, 5, 6, 7, 8, 9, 89, 11, 12, 1, 14, 15, 16, 1237, 18, 30, 20, 21, 22, 3234, 24, 3}Test No: 4 returns false for an unsorted table of size 5. This test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 5 | | Current Status: Passed | |
| Test tittle: incomplete table of size 5 | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize an unsorted table of size 5 and 24 values within. | Does system allow incomplete tables to exist | Error message |

Given the following values: {5, 2, 7, 4, 5, 6, 7, 8, 9, 89, 11, 12, 1, 14, 15, 16, 1237, 18, 30, 20, 21, 22, 3234, 24}, Test No: 5 returns the error message “ERROR!: Input must be square.” The test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 6 | | Current Status: Passed | |
| Test tittle: Excess elements for a table of size 5 | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize an unsorted table of size 5 and 26 values within. | Does system allow for an excess in elements in a table to exist | Error message |

Given the following values: {5, 2, 7, 4, 5, 6, 7, 8, 9, 89, 11, 12, 1, 14, 15, 16, 1237, 18, 30, 20, 21, 22, 3234, 24, 8434}, Test No: 6 returns the error message “ERROR!: Input must be square.” The test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 7 | | Current Status: Failed | |
| Test tittle: Sort an empty table with a size of 0 | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize an empty table of with a size of 0. | Initial condition | Creation of an empty table with a size of 0. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use sortable method to sort table. | Check that method accepts empty tables | A sorted empty table. |
| 4 | Use assertTrue to test that the isSorted method returned true. | Check that table is sorted | Test should pass |

Test No: 7 returns “Table is empty!!!!” instead of returning true and the sorted table. The test failed since it did not return true.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 8 | | Current Status: Passed | |
| Test tittle: Sort a table of size one | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a table of with a size of 1 containing 1 value. | Initial condition | Creation of a table with a size of 1. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use sortable to sortable method to sort the table. | Check that method accepts empty tables | A sorted table with 25 values. |
| 4 | Use assertTrue to test that the isSorted method returned true. | Check that table is sorted | Test should pass |

Given the values {5}, Test No: 8 returns true and the value within that is sorted. The test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 9 | | Current Status: Passed | |
| Test tittle: Sort an unsorted table of size 5 | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a table of with a size of 5 with 25 unsorted values within. | Initial condition | Creation of a table with a size of 5 and a total of 25 unsorted values. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use sortable method to sort table. | Check that method accepts empty tables | A table containing 25 sorted values. |
| 4 | Use assertTrue to test that the isSorted method returned true. | Check that table is sorted | Test should pass |

Given the unsorted values: {5, 2, 7, 4, 5, 6, 7, 8, 9, 89, 11, 12, 1, 14, 15, 16, 1237, 18, 30, 20, 21, 22, 3234, 24, 3}, Test No: 9 returns false for an unsorted table of size 5 and returns the 25 values within after sorting the table. The sorted table returned is: {1, 2, 3, 4, 5, 5, 6, 7, 7, 8, 9, 11, 12, 14, 15, 16, 18, 20, 21, 22, 24, 30, 89, 1237, 3234} After sorting the table, it returns true. This test passed.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No: 10 | | Current Status: Passed | |
| Test tittle: Sort a sorted table of size 5 | | | |
| Testing approach: This test will be conducted using the JUnit framework. | | | |
| STEP | Operator Action | Purpose | Expected Results |
| 1 | Initialize a table of with a size of 5 with 25 sorted values within. | Initial condition | Creation of an empty table with a size of 1. |
| 2 | Create an instance of a TableSorter using the default constructor. | Initial  condition | Creation of an instance of a TableSorter. |
| 3 | Use sortable method to sort table. | Check that method accepts empty tables | A table with 25 sorted values. |
| 4 | Use assertTrue to test that the isSorted method returned true. | Check that table is sorted | Test should pass |

Given the following table: {1, 2, 3, 4, 5, 6, 7, 8, 9. 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25}, Test No: 10 returns true for a sorted table of size 5 and returns the 25 values within sorted: {1, 2, 3, 4, 5, 6, 7, 8, 9. 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25}. The test passed.

80% of the tests passed when testing code 4478 using test plan 0538.