# Test Description

**Test Name or ID**: IT001

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the interaction between the Truck structure and the hasEnoughSpace function, ensuring that the function correctly evaluates whether the truck has enough capacity for a given shipment.

**Setup:** Initialize various Truck objects with specific volume and weight capacities.

**Test Function**: hasEnoughSpace

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid truck, enough space. | Truck {volCapacity=100, weightCapacity=200}, wght=50, vol=50 | 1 | 1 | Pass |
| Valid truck, not enough volume. | Truck {volCapacity=30, weightCapacity=200}, wght=50, vol=50 | 0 | 0 | Pass |
| Valid truck, not enough weight. | Truck {volCapacity=100, weightCapacity=40}, wght=50, vol=50 | 0 | 0 | Pass |
| Null truck. | Truck = NULL, wght=50, vol=50 | -1 | -1 | Pass |

**Bugs Found**: No bugs found during the test.

# Test Description

**Test Name or ID**: IT002

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the loadShipment function, ensuring that it correctly validates the box and weight, checks the truck's capacity, and updates the truck's capacity accordingly.

**Setup:** Initialize various Truck objects with specific volume and weight capacities.

**Test Function**: loadShipment

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid box size and weight, enough space. | Truck {volCapacity=100, weightCapacity=200}, wght=5, vol=1 | tr->volCapacity=99, tr->weightCapacity=195 | tr->volCapacity=99, tr->weightCapacity=195 | Pass |
| Invalid box size. | Truck {volCapacity=100, weightCapacity=200}, wght=50, vol=2 | "Box size is invalid!" | "Box size is invalid!" | Pass |
| Invalid weight size. | Truck {volCapacity=100, weightCapacity=200}, wght=-50, vol=1 | "Weight size is invalid!" | "Weight size is invalid!" | Pass |
| Not enough space in the Truck. | Truck {volCapacity=40, weightCapacity=30}, wght=50, vol=1 | "Truck is full!" | "Truck is full!" | Pass |

**Bugs Found**: No bugs found during the test.

# Test Description

**Test Name or ID**: IT003

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the checkMoreCapacity function, ensuring that it correctly compares the remaining capacities of two trucks and returns the correct result.

**Setup:** Initialize various Truck objects with specific volume and weight capacities.

**Test Function**: checkMoreCapacity

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| tr2 has more capacity. | Truck1 {volCapacity=50, weightCapacity=100}, Truck2 {volCapacity=85, weightCapacity=5} | 2 | 2 | Pass |
| tr1 has more capacity. | Truck1 {volCapacity=85, weightCapacity=5}, Truck2 {volCapacity=50, weightCapacity=100} | 1 | 1 | Pass |
| Null Truck1. | Truck1 = NULL, Truck2 {volCapacity=100, weightCapacity=200} | 2 | 2 | Pass |
| Null Truck2. | Truck1 {volCapacity=100, weightCapacity=200}, Truck2 = NULL | 1 | 1 | Pass |

**Bugs Found**: No bugs found during the test.

# Test Description

**Test Name or ID**: IT004

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the validateWeight function, ensuring that it correctly determines whether a given weight is valid.

**Setup:** Define various weights to test different scenarios.

**Test Function**: validateWeight

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid weight (positive). | weight=50 | 1 | 1 | Pass |
| Weight is zero. | weight=0 | 0 | 0 | Pass |
| Negative weight. | weight=-10 | 0 | 0 | Pass |
| Large positive weight. | weight=1000 | 1 | 1 | Pass |

**Bugs Found**:

No bugs found during the test.

# Test Description

**Test Name or ID**: IT005

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the validateBox function, ensuring that it correctly determines whether a given box size is valid based on predefined constants.

**Setup:** Define various box sizes to test different scenarios.

**Test Function**: validateBox

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid small box size. | size=SM\_BOX | 1 | 1 | Pass |
| Valid medium box size. | size=MD\_BOX | 1 | 1 | Pass |
| Valid large box size. | size=LG\_BOX | 1 | 1 | Pass |
| Invalid box size. | size=10 | 0 | 0 | Pass |

**Bugs Found**:

No bugs found during the test.

# Test Description

**Test Name or ID**: IT006

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the validateDest function, ensuring that it correctly determines whether a given destination point is within the map's bounds.

**Setup:** Define various Point objects with different row and column values to test different scenarios.

**Test Function**: validateDest

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid destination within bounds. | Point {row=5, col=5} | 1 | 1 | Pass |
| Invalid destination (negative row). | Point {row=-1, col=5} | 0 | 0 | Pass |
| Invalid destination (negative col). | Point {row=5, col=-1} | 0 | 0 | Pass |
| Invalid destination (exceed bounds) | Point {row=MAP\_ROWS, col=MAP\_COLS} | 0 | 0 | Pass |

**Bugs Found**: No bugs found during the test.

# Test Description

**Test Name or ID**: IT007

**Test Type**: Integration testing.

**Description**: These tests aim to integrate and validate the validateShipment function, ensuring that it correctly determines whether a given destination point is within the map's bounds.

**Setup:** Define various Point objects with different row and column values to test different scenarios.

**Test Function**: validateShipment

**Test Scenarios:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Test Data | Expected Result | Actual Result | Pass/Fail |
| Valid Shipment | {150, MD\_BOX, {10, 10}} | 1 | 1 | Pass |
| Invalid Box Size | {150, 7, {10, 10}} | 0 | 0 | Pass |
| Invalid Weight | {-10, MD\_BOX, {10, 10}} | 0 | 0 | Pass |
| Invalid Destination | {150, MD\_BOX, {-1, 50}} | 0 | 0 | Pass |

**Bugs Found**: No bugs found during the test.