Blackbox Test: Validate()

Function: Validate()	Data	Expected Result	Description
Test 1a	size: 0.50 weight: 500 struct Point des = {2, 2}	Return 1(user Inputs are validated)	Tests valid shipment input with a size of 0.50
Test 1b	size: 0.25 weight: 500 struct Point des = {2, 2}	Return 1(user Inputs are validated)	Tests valid shipment input with a size of 0.25
Test 1c	size: 1.00 weight: 500 struct Point des = {2, 2}	Return 1(user Inputs are validated)	Tests valid shipment input with a size of 1.00
Test 2a	size: -1.00 weight: 500 struct Point des = {2, 2}	Return -1(approriate error message is displayed for invalid size input)	Tests with invalid shipment size (negative size)
Test 2b	size: NULL weight: 500 struct Point des = {2, 2}	Return -1(approriate error message is displayed for invalid size input)	Test with NULL shipment size
Test 2c	size: 2.00 weight: 500 struct Point des = {2, 2}	Return -1	Test with invalid shipment size (over 1)

Test 2d	size: 0.5678 weight: 500 struct Point des = {2, 2}	Return -1	Test with invalid shipment size of 0.5678
Test 2e	size: 1234567890 weight: 500 struct Point des = {2, 2}	Return -1	Test with invalid shipment size of 1234567890
Test 3a	size: 0.50 weight: 1 struct Point des = {2, 2}	1	Tests with valid shipment weight (min)
Test 3b	size: 0.50 weight: 1000 struct Point des = {2, 2}	1	Tests with valid shipment weight (max)
Test 3c	size: 0.50 weight: 300 struct Point des = {2, 2}	1	Tests with valid shipment weight
Test 4a	size: 0.50 weight: 0 struct Point des = {2, 2}	-2	Tests with invalid shipment weight (0 – under min)
Test 4b	size: 0.50 weight: 100000000 struct Point des = {2, 2}	-2	Tests with invalid shipment weight (over max)

Test 4c	size: 0.50 weight: NULL struct Point des = {2, 2}	-2	Test with invalid shipment weight (NULL)
Test 4d	size: 0.50 weight: -1 struct Point des = {2, 2}	-2	Tests with invalid shipment weight (negative number)
Test 4e	size: 0.50 weight: 1001 struct Point des = {2, 2}	-2	Test with invalid shipment weight (max + 1)
Test 5a	size: 0.50 weight: 1 struct Point des = {0,1}	1	Tests with valid destination input (min)
Test 5b	size: 0.50 weight: 1 struct Point des = {25, 25}	1	Tests with valid destination input (max)
Test 5c	size: 0.50 weight: 1 struct Point des = {2, 2}	1	Tests with valid destination input
Test 5d	size: 0.50 weight: 1 struct Point des = {8, 24}	1	Tests with valid destination input

Test 6a	size: 0.50 weight: 1 struct Point des = {NULL, 1}	-3	Test with invalid destination input (NULL)
Test 6b	size: 0.50 weight: 1 struct Point des = {25, NULL}	-3	Test with invalid destination input (NULL)
Test 6c	size: 0.50 weight: 1 struct Point des = {0, 26}	-3	Test with invalid destination input (max + 1)
Test 6d	size: 0.50 weight: 1 struct Point des = {26, 0}	-3	Test with invalid destination input (max + 1)
Test 6e	size: 0.50 weight: 1 struct Point des = {0,0}	-3	Test with invalid destination input (0,0 is start)
Test 6f	size: 0.50 weight: 1 struct Point des = {10, 50}	-3	Test with invalid destination input (over max)
Test 6g	size: 0.50 weight: 1 struct Point des = {-2, 20}	-3	Test with invalid destination input (negative number)
Test 6h	size: 0.50 weight: 1 struct Point des = {50, 10}	-3	Test with invalid destination input (over max)
Test 6i	size: 0.50 weight: 1 struct Point des = {2, -1}	-3	Test with invalid destination input (negative number)
Test 6j	size: 0.50 weight: 1 struct Point des = {-1, -1}	-3	Test with invalid destination input (both negative number)

Test 7a	size: ‘B’ weight: 10 struct Point des = {10, 10}	-1	Test with invalid size (special character)
Test 7a	size: 0.50 weight: ‘@’ struct Point des = {10, 10}	-2	Test with invalid weight (special character)
Test 7a	size: 0.50 weight: 10 struct Point des = {A, Z}	-3	Test with invalid destination input (both special character)