Acceptance Test

The acceptance tests will follow the same testing style as the integration tests but will specifically target and test for all the business requirements.

Acceptance Test #	Business Req # Focused	Description	Test Data	Expected Result

T01a	R001	Tests with all valid variables and package destination closest to green truck.
		The package size is 0.50
			init()
			struct Shipment package = {10, 0.5, {7, 24}}
				4

T01b	R001	Tests with all valid variables and package destination closest to green truck
		The package size is 0.25
			init()
			struct Shipment package = {10, 0.25, {7, 24}}
				4

T01c	R001	Tests with all valid variables and package destination closest to green truck
		The package size is 1
			init()
			struct Shipment package = {10, 1, {7, 24}}
				4

T01d	R001	Tests with all valid variables, except invalid package size and package destination closest to green truck.
		The package size is 0.75 (invalid)
			init()
			struct Shipment package = {10, 0.75, {7, 24}}
				-1

T02a	R002	Tests with all valid variables and package destination closest to green truck
		The package weight is 999
			Init()
			struct Shipment Package = {1000, 0.5, {7,24}}
				4

T02b	R002	Tests with all valid variables and package destination closest to green truck
		The package weight is 1
			Init()
			struct Shipment Package = {1000, 0.5, {7,24}}
				4

T02c	R002	Tests with all valid variables and package destination closest to green truck
		The package weight is 0
			Init()
			struct Shipment Package = {1000, 0.5, {7,24}}
				-1

T02d	R002	Tests with all valid variables and package destination closest to green truck
		The package weight is 1001
			Init()
			struct Shipment Package = {1000, 0.5, {7,24}}
				-1

T03a	R003	Tests with all valid variables and package destination closest to blue truck	Init()
			struct Shipment Package = {10, 0.5, {11,11}}
				2

T03b	R003	Tests with all valid variables, except invalid package destination and package destination closest to blue truck	Init()
			struct Shipment Package = {10, 0.5, {11,11}}
				-1

T04	R004	Tests with all valid variables and package destination closest to green truck	Init()
			struct Shipment Package = {10, 0.5, {7,24}}
				4

T05	R005	Tests with all valid variables and package destination closest to green truck.
		However green truck would be over capacity if it accepts the package, and blue truck is 2nd closest to destination.
			Init()
			struct Shipment Package = {510, 0.5, {7,24 }}
			trucks[1].cargoWeight = 500
				2

T06	R006	Tests with all valid variables and package destination closest to green truck.
		However, green would be over the volume capacity if it accepts the package, and blue truck is the 2nd closest to the destination.
			Init()
			struct Shipment Package = {510, 1, {7,24}}
			trucks[1].cargoVol = 35.5
				2

T07	R007	Tests with all valid variables and package destination closest to green truck.
		Also green truck requires 0 divergences with the tested destination.
			Init()
			struct Shipment Package = {10, 0.5, {7,24}}
				4

T08	R008	Tests with all valid variables and package destination closest to green truck.
		Tests to see if blue will take priority instead of the green truck given the weight capacity differences.
			Init()
			struct Shipment Package = {510, 0.5, {7,23 }}
			
			trucks[1].cargoWeight = 500
				2

T08	R009	Tests with all valid variables and package destination closest to green truck.
		Tests to see if blue will take priority instead of the green truck given the volume capacity differences.
			Init()
			struct Shipment Package = {510, 1, {7,23}}
			
			trucks[1].cargoVol = 35.5
				2

T10	R010	At destination {1, 1} all trucks have the same distance from the package.
		All trucks also have equal weight and volume.
		The blue truck should be prioritized in this case.
			Init()
			struct Shipment Package = {10, 0.5, {1,1}}
				2

T11	R011	At destination {1, 1} all trucks have the same distance from the package.
		
		However, all trucks are too full to accept the package for today.
			Init()
			struct Shipment Package = {10, 0.5, {1,1}}
				0