# Function Description

**Function Name:** isTruckOverloaded

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| currentLoad | Float | The current shipment weight of the truck |
| shipment\_weight | Float | The weight of the new shipment that will be loaded to the truck |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** An integer to indicate if the truck will be overloaded when the new shipment is added. Returns 1 if overloaded, otherwise it will return 0.

**Description:** Checks whether the shipment weight will exceed the capacity of the truck if the new shipment is added to the existing cargo weight. This will help determine if the truck is able to add more cargo to the truck.

**Function Name:** isBoxSizeExceeded

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| currentBoxSize | Float | The current box size in cubic meters that is already in the truck |
| boxSize | Float | The box size of the incoming shipment |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** An integer representing the boolean value of the truck’s capacity for loading box sizes. Returns 1 if the total box size exceeds the capacity of the truck.

**Description:** Checks if the sum of the current box size and the incoming box size will exceed the capacity of the truck. If the sum of the two box sizes exceeds 36 cubic meters it will return 1, otherwise it will return 0. This will help determine if the truck is capable of loading another box.

**Function Name:** vaildCargo

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| boxSize | Float | The box size of the incoming shipment |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** An integer representing the boolean value of the incoming cargo box size. Return 1 if the box size is not valid.

**Description:** Checks if the incoming box is 0.25, 0.5 or 1 cubic meters. If the box is not one of the standard boxes it returns 1, otherwise it will return 0. This will help determine if the box is valid.

**Function Name:** findValidTruckPaths

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| shipment | Struct | Shipment struct containing the destination coordinates (row and column) of the shipment. |
| truck | Struct | Truck struct containing the route the truck travels on the map. |
| map | Struct | Map struct containing all the points (squares) in the map. |
| Routes | Array | Array of pointers to Route structs to store the valid routes |
| Size | Int | Pointer to an integer to store the number of valid routes. |
|  |  |  |

**Returns:** void, populates the "routes" array and updates the "size" variable

**Description:** Populates the "routes" array with routes that reach the given shipment destination.

**Function Name:** hasDestination

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| Routes | Array | Pointer to a Route struct representing a calculated route. |
| Shipment | Struct | Shipment struct containing the destination coordinates (row and column) of the shipment. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** int, 1 if the destination is in the route, 0 if not

**Description:** Checks if the given route contains the destination coordinates of the shipment.

**Function Name:** printRoute

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| Routes | Array | Route struct containing the points (coordinates) of the route. |
| Shipment | Struct | Shipment struct containing the destination coordinates (row and column) of the shipment. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** void, does not return a value, only prints to the console.

**Description:** Prints the points (coordinates) of a route to the console.

**Function Name:** isBuildingIntersected

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| Routes | Array | Route struct to check for intersections with buildings. |
| Map | Struct | Map struct containing the buildings on the map. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** int, 1 if the route intersects with a building, 0 if not

**Description:** Checks if a given route intersects with any buildings on the map.