# Function Description

**Function Name:** FindClosestTruck

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| trucks[] | struct Truck | an array of trucks that holds the blue, yellow, and green trucks. |
| numOfTrucks | const unsigned int | holds the number of trucks currently stored in the trucks array. |
| desination | const struct Point\* | holds the destination of the package. |

**Returns:** an integer type that represents the index of the closest truck. If the destination supplied in the argument is invalid, then it returns -1.

**Description:** Finds the closest truck to the destination of the package.

**Function Name:** FindTruckForShipment

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| trucks[] | struct Truck | an array of trucks that holds the blue, yellow, and green trucks. |
| numOfTrucks | const unsigned int | holds the number of trucks currently stored in the trucks array. |
| destination | const Struct Point\* | holds the destination of the package. |
| package | const struct Box\* | a data structure of type Box that contains the weight and size of the package. |

**Returns:** an integer type that represents the index of the truck that will be used for shipping the package. If the truck cannot ship the package because of an invalid destination or the package is too big for the truck then return -1.

**Description:** Find the best truck for shipping the package. It considers looking for the closest truck to the destination and the truck that has the most space which can fit the package.

**Function Name:** addPackageToTruck

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| truck | struct Truck\* | holds a reference to the truck that will ship the package. |
| package | const struct Box\* | holds the package that will be shipped. |

**Returns:** an integer type that represents the successful loading of the package. If the package has been successfully loaded onto the truck, then return 0, otherwise return -1.

**Description:** Load the package to the truck. Before loading, it will first check whether the package can fit into the truck. If the package can fit into the truck, then add the package to the truck.

**Function Name:** IsValidSize

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| size | double | holds the size of the package entered by the client. |

**Returns:** an integer type that represents successful validation. If the size/volume entered by the client is within the range of valid sizes/volumes then return 0, otherwise -1.

**Description:** Validate the size entered by the client. This will be called before creating a Box object to validate user input data.

**Function Name:** IsValidWeight

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| weight | unsigned int | holds the weight of the package entered by the user. |

**Returns:** an integer type that represents successful validation. If the weight entered by the client is less than the maximum load that a truck can hold then return 0, otherwise -1.

**Description:** Validate the weight entered by the client. This will be called before creating a Box object to validate user input data.

**Function Name:** IsValidDestination

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| destination | const struct Point\* | holds a reference to the destination entered by the client. |

**Returns:** an integer type that represents successful validation. Check if the row and column entered by the client are valid. If the row is greater than 0 and less than or equal to 25 then the row is valid. It then checks if the column is greater than or equal to ‘A’ and less than or equal to ‘Y’, then it is a valid column. If both row and column are valid then return 0, otherwise -1.

**Description:** Validate the weight entered by the client. This will be called before creating a Box object to validate user input data.