# SFT PROJECT - MILESTONE 3

## **SECTION: ZBB**

## **GROUP: D**

## **Professor: Kaveh Eshraghian**

## **TEAM MEMBERS: ARSHNOOR KAUR, GURMEHAK KAUR UPPAL, AJAYPARTAP SINGH MAAN, MANAS GANDOTRA**

## **FUNCTION DESCRIPTIONS**

1. **Function Name: assignShipmentToTruck**

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| deliveryMap | const struct Map\* | Pointer to the map of the delivery area with buildings on it. |
| fleet | struct Truck[] | |  | | --- | | Array of trucks including the route for each truck. |  |  | | --- | |  | |
| totalTrucks | int | The number of trucks in the array. |
| shipmentDetails | const struct PackageInfo\* | Pointer to the shipment details including size and weight. |

**Returns:** ‘int’ - The index of the truck in the array. If no truck can take the shipment, -1 is returned.

**Description:** Assigns a shipment to the best available truck considering load, size, weight, and route. Iterates through the fleet of trucks, checks capacity, and finds the nearest truck. If suitable, adds the shipment and updates the truck's capacity.

1. **Function Name:** isTruckCapacitySufficient

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| truck | const struct Trucks\* | |  | | --- | | Pointer to the truck being checked. |  |  | | --- | |  | |
| shipment | const struct PackageInfo\* | Pointer to the shipment details including size and weight. |

**Returns:** ‘int’ - Returns 1 if the truck can hold the shipment, otherwise 0.

**Description:** Checks if the truck has sufficient capacity to hold the shipment by comparing the truck's available weight and volume with the shipment's weight and size.

1. **Function Name:** displayRouteToDestination

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| route | const struct Route\* | |  | | --- | | Pointer to the route followed by the truck. |  |  | | --- | |  | |
| destination | |  | | --- | | const struct Point |  |  | | --- | |  | | The delivery destination point. |

**Returns:** void - No return value.

**Description:** Displays the route that the truck used to deliver the package to its destination by iterating through the points in the route.

1. **Function Name:** validateShipment

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| shipment | const struct PackageInfo\* | Pointer to the shipment details including size and weight. |

**Returns: int - Returns 1 if the shipment details are valid (weight > 0, size is 1, 3, or 5 cubic meters, and destination is within valid range), otherwise 0.**

**Description:** Validates the shipment details by checking if the weight is greater than 0, the size is one of the allowed values (1, 3, or 5 cubic meters), and the destination point (row and column) falls within the acceptable range. If any detail is invalid, an appropriate error message is printed and the function returns 0.

1. **Function Name:** **updateTruckCapacity**

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| truck | struct Truck\* | Pointer to the truck being updated. |
| shipment | const struct PackageInfo\* | Pointer to the shipment details including size and weight. |

**Returns:** void - No return value.

**Description:** Updates the truck's available weight and volume after adding the shipment by subtracting the shipment's weight and size.

1. **Function Name:** findNearestTruck

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| map | const Struct Map\* | Pointer to the map of the delivery area with buildings. |
| fleet | struct Truck\* | Array of Truck structures representing the fleet. |
| totalTrucks | int | Number of trucks in the fleet array. |
| pickupLocation | const struct Point\* | Location where the shipment needs to be picked up. |

**Returns:** int - The index of the nearest truck in the array. If no truck is available, -1 is returned.

**Description:** Finds the nearest available truck to the shipment's pickup location by iterating through the fleet and calculating the distance between each truck's current location and the pickup location.

1. **Function Name:** initialiseTruck

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| truckID | int | The unique ID for the truck. |
| route | char | The route character for the truck (B, Y, G). |

**Returns:** struct Truck\* - A pointer to the initialized Truck structure.

**Description:** Initializes a Truck structure with default values, including setting available weight to 2500 kg, available volume to 100 cubic meters, and setting the packages pointer to NULL. It also sets the truckID and route based on the provided parameters.

1. **Function Name:** initializePackage

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| (none) | (none) |  |

**Returns:** struct PackageInfo\* - A pointer to the initialized PackageInfo structure.

**Description:** Initializes a PackageInfo structure by prompting the user to enter the weight, size, and destination of the package. Validates the input using the validateShipment function to ensure all details are correct.