# Function Description

**Function Name:** findTruckForShipment

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| map | struct Map | The map of the delivery area with buildings on it. |
| trucks | struct TruckCapacity[] | An array of trucks including the route for each of the trucks. |
| numTrucks | int | The number of trucks in the array of trucks. |
| Shipment | int | A data structure containing the size and weight of the shipment. |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** An integer representing the index of the truck in the trucks array on which the shipment should be placed. If no truck can take the shipment, then -1 is returned.

**Description:** Finds the best truck for a shipment by considering the load on the truck, the size and weight of the shipment, and the route of the truck to try to place it on a truck which goes closest to the destination.

**Function Name:** calculateDistance

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| X1 | int | The X-coordinate of the first point |
| Y1 | int | The Y-coordinate of the first point |
| X2 | int | The X-coordinate of the second point |
| Y2 | int | The Y-coordinate of the second point |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** The Euclidean distance between the two points.

**Description:** Calculates the Euclidean distance between two points (x1, y1) and (x2, y2) on a 2D plane.

**Function Name:** determineTruckRoute

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| map | struct Map | The map of the delivery area with buildings on it. |
| start | struct Point | The starting point of the truck. |
| destination | struct Point | The destination point for the delivery. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** A list of coordinates representing the route from the starting point to the destination.

**Description:** Determines the route for a truck based on the current map and the destination. It uses a pathfinding algorithm to find the shortest path that avoids obstacles (buildings).

**Function Name:** calculateTruckLoad

**Parameter List:**

|  |  |  |
| --- | --- | --- |
| Parameter Name | Type | Description |
| Truck | struct TruckCapacity | The truck for which the load is being calculated. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Returns:** A data structure containing the current weight and volume of the truck.

**Description:** Calculates the current load (weight and volume) of a truck based on the shipments it carries.