TRAFFIC DRONE CONTROLLER

ELEMENTARY DATA STRUCTURES AND LOGICAL THINKING ASSIGNMENT

Name: Keerthan S
Roll Number ME24B1006
Course Code: CS1002

PROBLEM STATEMENT

The growing use of cargo drones in urban and rural logistics demands a system to manage deliveries efficiently. This assignment simulates a Cargo Drone Traffic Controller that manages delivery requests, urgent dispatches, flight logs, overloaded drones, and emergency reroutes. The simulation uses appropriate data structures to mimic real-time drone traffic behavior.

OBJECTIVES

- Simulate delivery request handling using a queue (FIFO).
- Dispatch urgent deliveries using a stack (LIFO).
- Maintain a flight log with a fixed-size array.
- Track overloaded drones using a singly linked list.
- Track serviced drones using a doubly linked list.
- Handle emergency reroutes using a circular linked list.
- Create a menu-driven interactive interface for users to manage drone tasks.

03

DESIGN EXPLANATION

Objective	Data Structure	Reason
Delivery requests	Queue (Array-based)	First-come-first-served delivery simulation
Urgent dispatches	Stack (Array-based)	Last-in-first-out ensures latest critical tasks are dispatched first
Flight log	Fixed-size array	Limited memory to simulate onboard flight history
Overloaded drones	Singly linked list	Simple insertion at the head, minimal traversal
Serviced drones	Doubly linked list	Allows forward and backward inspection of service records
Emergency rerouting	Circular linked list	Continuous routing and cyclic priority management

04

LOGIC OF THE CODE

User Interface:

- A menu allows the user to select from different drone management tasks like enqueueing, dispatching, logging, etc.

Delivery Request System (Queue):

- Enqueue items like "Food", "Medicine", etc.
- FIFO logic ensures oldest requests are handled first.

Urgent Dispatch (Stack):

- Dequeued requests are pushed into a stack.
- Stack ensures LIFO behavior, dispatching the most recent urgent task.

Flight Log (Array):

- Logs completed deliveries in a fixed-size array.
- When full, it removes the oldest entry (FIFO-style rolling log).

Overloaded Drone Tracker (Singly Linked List):

- Drones that carry too much weight are inserted at the head.
- Allows quick insertion and simple management.

Serviced Drones (Doubly Linked List):

- Once serviced, drones are moved from overloaded list into a doubly linked list.
- Can be viewed in both forward and backward directions.

Emergency Reroutes (Circular Linked List):

- Drones needing urgent re-routing are added to a circular list.
- The list is traversed twice to simulate continuous reroute checks.

05

VARIABLES AND FUNCTIONS USED

Key Variables:

- queue, stack, flightLog: Arrays for basic operations
- f, r, top, logCount: Pointers/counters
- overhead, servhead, tail: Pointers to linked list heads

Key Functions:

Function	Purpose
enqueue()	Adds delivery request to queue
dequeue()	Removes item from front of queue
push() / pop()	Stack operations for urgent dispatch
logDelivery()	Adds delivery to rolling flight log
insertOverloaded()	Inserts a drone into overloaded list
moveToServiced()	Moves drone to serviced (doubly linked) list
traverseServicedForward() / Backward()	Views serviced drones
insertEmergency()	Adds drone to emergency circular list
traverseEmergencyTwice()	Simulates 2-cycle rerouting

06

SAMPLE OUTPUT

```
Cargo Drone Traffic Controller
1. Enqueue Delivery Request
2. Dispatch Drones (LIFO)
3. Log Delivery
4. Add Overloaded Drone
5. Move Drone to Serviced
6. Show Serviced Drones (Forward)
7. Show Serviced Drones (Backward)
8. Add Emergency Reroute Drone
9. Show Emergency Reroutes (2 Cycles)
10. Show Flight Log
0. Exit
Enter choice: 1
Enter delivery item: Food
```

```
Enter choice: 1
Enter delivery item: Water
```

```
Enter choice: 1
Enter delivery item: Kite
```

```
Enter choice: 2
Dispatching (LIFO): Kite -> Water -> Food -> NULL
```

```
Enter choice: 2
Dispatching (LIFO): NULL
```