# **ELEMENTARY DSA ASSIGNMENT REPORT**

**Cargo Drone Traffic Controller**

Name: Konda Nihaal Sekhar

Roll Number: ME24B1062

**What problem are you solving?**

The problem statement is to design a Cargo Drone Traffic Controller system that handles delivery requests, urgent dispatches, flight logs, and drone maintenance using appropriate data structures such as Queue, Stack, Array, Singly Linked List, Doubly Linked List, and Circular Linked List.

**Key Objective:**

The main objective is to simulate drone traffic management, logging, and maintenance.

**Why did you choose specific data structures?**

The data structures used for each task given are: -

* Queue (Array): Ground requests arrive in order — fits naturally with FIFO (First-In-First-Out).
* Stack (Array): Urgent tasks are handled by LIFO (Last-In-First-Out).
* Circular Array: Saves memory, keeps only the latest 6 logs efficiently.
* Singly Linked List: Simple insert/remove for overloads.
* Doubly Linked List: Allows both forward and backward traversal for diagnostics.
* Circular Linked List: Ideal for emergency drone rerouting in loops.

**Explain how your code works step by step :**

**Request Dispatch**:

* Enqueue requests.
* Dequeue and push to stack (LIFO).
* Pop and dispatch in reverse order (urgent first).

**Flight Logging**:

* Store latest 6 delivery records in a circular manner.
* Oldest entries are overwritten.

**Overload Drones**:

* Insert overloaded drones into a singly linked list.
* Remove them once serviced.

**Service Diagnostics**:

* Insert into head of doubly linked list.
* Traverse forward and backward for service history.

**Emergency Reroute**:

* Insert drones into a circular linked list.
* Loop through them infinitely as needed.

**Bonus :**

**(a) Urgent Delivery Priority – LIFO Justification**

LIFO (Last-In, First-Out) is used for urgent deliveries, such as "Fuel," which might be requested last minute but need quick dispatch. A stack ensures that the most recent and critical deliveries are prioritized.

**(b) Flight Log Overflow – Archiving Justification**

Flight logs are stored in a fixed-size array. When full, older entries are archived, similar to how real aviation systems move old data to long-term storage for easier access to recent deliveries.

**(c) Drone Overload – Cause and Resolution**

"Drone3" became overloaded due to excess weight. It was serviced and recalibrated for better payload handling, then moved to a serviced tracker to avoid future issues.

**(d) Emergency Reroute – Design Justification**

"Drone4" encountered a storm and needed to reroute. A circular linked list helps continually assess alternative paths, ensuring the drone stays in safe airspace until it can resume its route.

**Variables :**

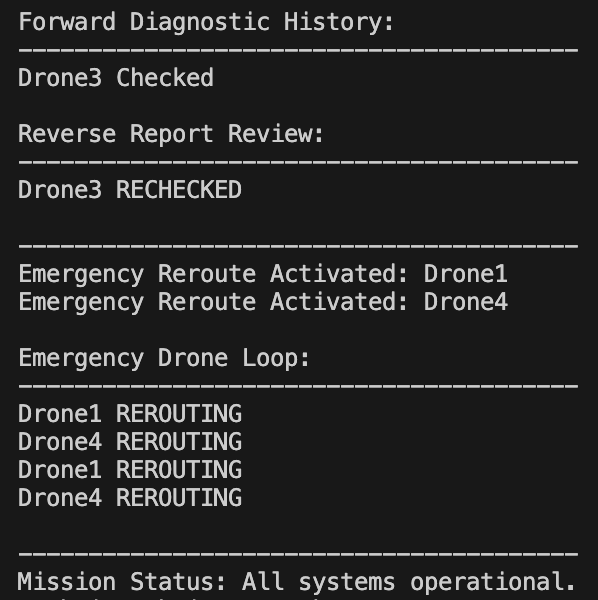
|  |  |
| --- | --- |
| *Variables* | *Type* |
| queue[] | Holds the incoming requests |
| stack[] | Prioritizes requests for urgent dispatch |
| arr[] | Stores deliver log |
| overload\_head | Head of single linked list for overload |
| service\_head | Head for double linked list for service |
| emergency\_tail | Tail for circular linked list for emergency |

**Functions :**

|  |  |
| --- | --- |
| *Function* | *Type* |
| enqueue() | Adds new request to the queue |
| dequeue() | Removes and returns oldest request |
| push()/pop() | Stack operation for LIFO dispatch |
| log\_delivery() | Log each delivery in circular array |
| insert\_overload() / remove\_overload() | Manage drones reporting overload using single linked list |
| insert\_service() / traverse\_service() | Track and review diagnostic history using double linked list |
| insert\_emergency() / traverse\_emergency() | Handle emergency rerouting using circular linked list |

**Output :**

****

****

**Conclusion :**

The simulation shows how data structures can be used to manage a complex drone logistics system. Each part of the system helps keep things running smoothly, whether everything is going as planned or in an emergency. Using linked lists and circular structures makes the system more flexible and easier to adapt to changing situations.