**Project Proposal: Hotel Management System**

**Important Points**

* The stated problem can be solved individually or in a group of two.
* This assignment will follow viva and the performance in viva is key for good evaluation.
* The assignment must be submitted on or before the deadline. Late submission will have ZERO grade.

**Project Description:**

You are tasked with developing a **Hotel Management System** that uses three data structures to efficiently manage operations. The system must handle floor and room allocation, manage booking requests, and keep a history of operations for audit and rollback purposes. Below is the scenario:

**Scenario**

The **Galaxy Hotel** has the following management requirements:

1. **Floor and Room Management**:
   * The hotel has F floors, with each floor containing a fixed number of residence units (N).
   * Each residence unit is either a Single Room, a Double Room or a Suit. (**type**)
   * Each residence unit has a unit identification number (**ID**)
   * Each residence unit can have only following status:
     1. Occupied (The customer has arrived and living in this unit)
     2. Booked (Booked for future dates)
     3. Ready (Ready to be booked or to occupy)
     4. Unavailable (Not available due to some reason)
2. **Booking Requests**:
   * Customers send booking requests specifying their desired room type and the number of nights they want to stay.
   * The booking request can only be entertained for next 30 days. (availability status for a room must be maintained for next 30 days)
   * There are some especial requests (HIGH PRIORITY REQUESTS) that are to be entertained with HIGH PRIORITY i.e., they are entertained before regular requests.
   * Every regular booking request is added to the queue.
   * Once ten requests are added to the queue, they are processed.
   * High Priority Requests are handled before regular requests
   * FIFO Order is maintained between requests of same type
3. **Booking and Operation History**:
   * Every successful booking must be recorded, including details like customer name, room number, and stay duration.
   * If a booking is canceled, the most recent booking for the specified room should be removed.
   * Use a **stack data structure** to store the history of bookings for auditing purposes, supporting rollback and review operations.

**Question Statement**

Design and implement a **Hotel Management System** using the following data structures:

1. **Tree Data Structure (For Bonus Marks)**:
   * Design a tree to represent the hotel's floors and rooms.
   * Ensure the tree can store information about each room’s type (Single, Double, or Suite) and allow for quick retrieval based on room type.
2. **Link List:**
   * Design Complete management system, manage floors, rooms, type of room.
3. **Queue Data Structure**:
   * Implement a queue to handle booking requests.
   * The system must process the requests in a First-In-First-Out (FIFO) manner.
4. **Stack Data Structure**:
   * Use a stack to maintain the booking history.
   * The stack must support operations like push (record new booking) and pop (remove the most recent booking).

**Example Operations:**

1. **Floor and Room Management**:
   * Create a data structure for a 5-floor hotel with 10 rooms on each floor, assigning room types.
   * Retrieve data like how many rooms are booked.
   * Which type of rooms are available.
2. **Handle Booking Requests**:
   * Enqueue the following requests in the system:
     + Request 1: Single room for 3 nights by Ahmad.
     + Request 2: Suite room for 2 nights by Ali.
   * Deque the request.
   * Display the current status of the queue stack after processing a batch of booking requests.
3. **Record Booking History**:
   * Add successful bookings to the stack.
   * Cancel the most recent booking and roll back the change.
   * Display the current status of the stack after processing a batch of booking requests