

Data Structures and Algorithms Project: Flight Reservation System

Course: Data Structures and Algorithms

Teaching Assistant: Muhammed Batuhan Karakoç

Email: mbatuhankarakoc@erciyes.edu.tr

Office Hours: Thursday, 09:30 - 11:45

Project Overview

The objective of this project is to design and implement a console-based application in Java that solves a given problem by effectively utilizing appropriate data structures and algorithms. The application will be controlled entirely through text commands entered into the console (CLI). Your program's main function should read these commands, parse them, and call the appropriate Java methods to execute the actions.

The project must be written in Java as a standalone, CLI-based program by teams of two. No external databases or GUIs are to be used. You are permitted to use Java's standard data structure implementations from the `java.util` package.

Project Details: Flight Reservation System

This project's goal is to create a flight reservation system incorporating the use of heaps, hash maps, and queues.

Core Requirements:

1. **Flight Management:** Store flight information (flight number, origin, destination, seats) in a hash map with the flight number as the key for fast lookup.
2. **Ticket Booking & Waitlisting:** Users can book tickets. Store bookings in a priority queue based on the booking time. If a flight is fully booked, use a queue to manage the waitlist on a first-come-first-served basis.
3. **Flight Status Management:** Implement a function to update a flight's status (e.g., `On Time`, `Delayed`, `Cancelled`). If a flight's status is updated to `Cancelled`, all existing bookings for that flight must be automatically removed, and its waitlist must be cleared.

Required Data Structures:

- **Hash Map:** For storing flight information based on flight numbers.

- **Priority Queue:** To manage ticket bookings based on the booking time.
- **Queue:** To manage waitlisted passengers for fully booked flights.

Required CLI Commands:

- `add_flight <FlightNum> <Origin> <Destination> <Seats>`
- `book_ticket <PassengerName> <FlightNum>`
- `cancel_booking <PassengerName> <FlightNum>`
- `update_flight_status <FlightNum> <NewStatus>`
- `view_flight_details <FlightNum>`

Submission and Evaluation

Your final submission must be a single `.zip` file containing all your `.java` source files. The project evaluation will be conducted through a final presentation and a Q&A session where you will demonstrate your application and explain your design choices.