PROJECT REPORT

AIRLINE MANAGEMENT SYSTEM

1. TITLE PAGE

Title*:* *Airline Management System*
Course: —Btech in Computer Science & Engineering
Submitted By: —Payal Rathod(590028512) & Palak Singh(590028490)
Faculty: —Mr. Vinod Kumar

2. ABSTRACT

A console-based software program called the Airline Reservation System was created to streamline and automate the airline ticket booking process. The goal of this project is to create an effective, user-friendly system that enables travellers to view available flights, enter personal information, and confirm their reservations via an organized, interactive interface. To ensure a seamless booking experience, the system incorporates crucial features like passenger data handling, flight menu display, reservation confirmation, and ticket generation. The project illustrates the useful application of data structures, functions, file handling concepts, and modular programming techniques using the C programming language. This project's main goals are to minimize errors, lessen manual labour, and offer a quick and dependable digital substitute for conventional ticket booking techniques. All things considered, this system highlights the significance of structured programming and logical system design in resolving real-world issues while providing a fundamental model for comprehending how actual airline reservation platforms function.

3. PROBLEM DEFINITION

The services offered by airlines include ticket booking, passenger check-in, and luggage management. These, when performed manually, may result in:

- Errors on passenger data
- Slow check-in process
- Difficulty in consistently managing baggage handling
- Time-consuming updates

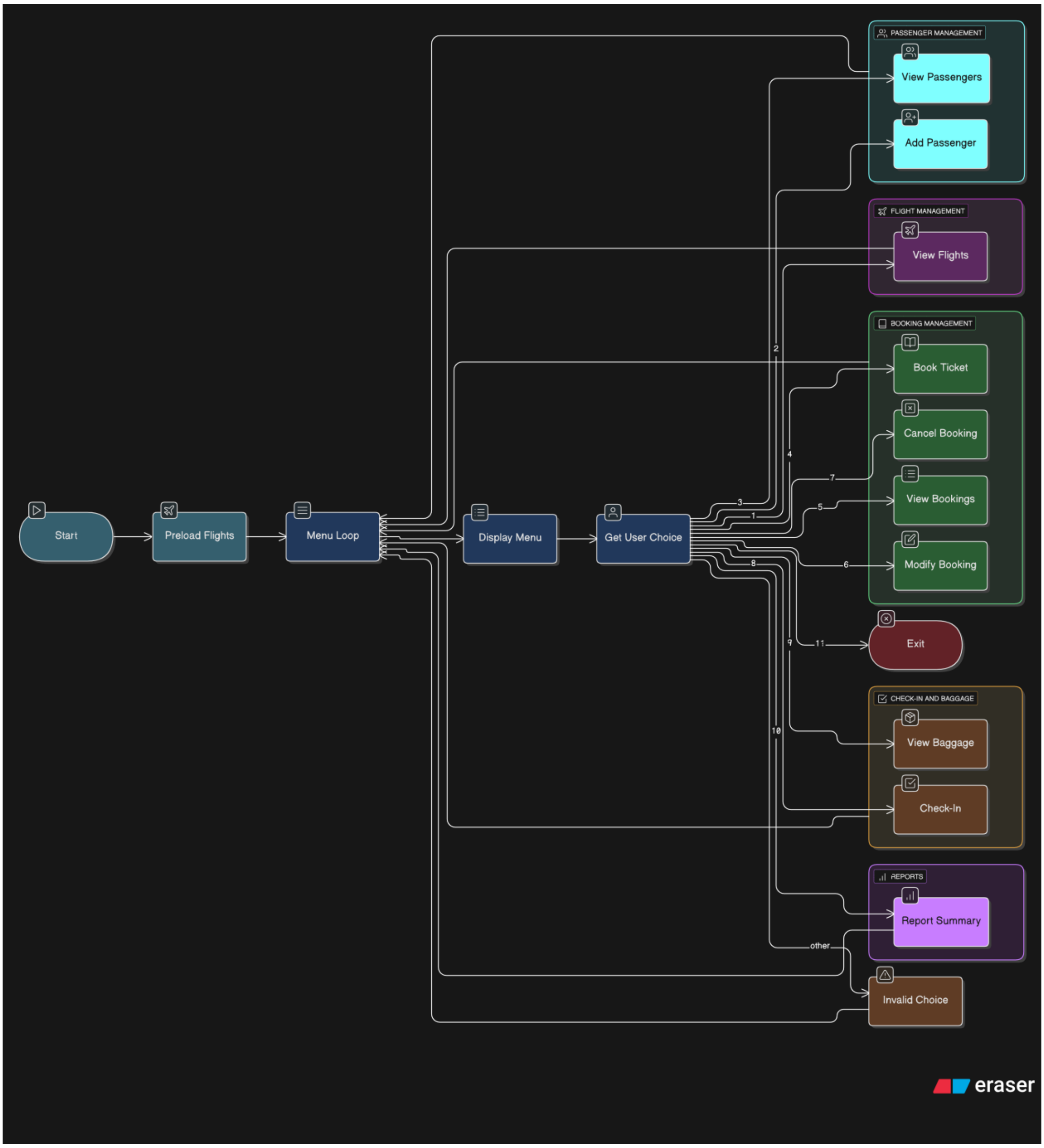
This project provides a single system that:

- Flight ticket booking
- Passenger check-in
- Tracks baggage weight and counts
- Results in structured output

---

4. SYSTEM DESIGN

4.1 FLOWCHARTS WITH SHAPES



4.2 ALGORITHMS

Algorithm 1: Ticket Booking

```
1. Start
2. Ask user for passenger details (name, age, flight, date)
3. Store all details in a structure array
4. Increase booking count
5. Display confirmation
6. Stop
```

Algorithm 2: Check-In

```
1. Start
2. Read booking number
3. Search booking record
4. If found → mark as checked-in
5. Else → show “record not found”
6. Stop
```

Algorithm 3: Baggage Entry

```
1. Start
2. Ask for passenger ID
3. Validate passenger exists
4. Ask number of bags and weight
5. Store in baggage structure
6. Print baggage slip
7. Stop
```

---

5. IMPLEMENTATION DETAILS (WITH SNIPPETS)

5.1 Structure for Ticket Booking

```
struct reservation {
    char name[50];
    int age;
    char flight[20];
    char date[15];
    int booking_id;
};
```

5.2 Structure for Check-In

```
struct checkin {
    int booking_id;
    int checked_in;
};
```

5.3 Structure for Baggage

```
struct baggage {
    int booking_id;
    int bag_count;
    float total_weight;
};
```

5.4 Booking Function

```
void bookTicket() {
    printf("Enter Name: ");
    scanf("%s", r[count].name);
    printf("Enter Flight: ");
    scanf("%s", r[count].flight);
    r[count].booking_id = count + 1;
    printf("Booking Successful! ID = %d", r[count].booking_id);
}
```

6. TESTING & RESULTS

Test Case 1 – Booking

Input	Expected Output
Name: Rahul	Booking confirmed
Flight: AI202	ID generated

Test Case 2 – Check-In

Booking ID	Expected Result
1	Check-in success
50	Not found

Test Case 3 – Baggage

Bags	Weight	Expected Output
2	18kg	Slip generated

The system passed all major functional tests.

7. CONCLUSION & FUTURE WORK

In conclusion

The project effectively replicates the booking, check-in, and baggage tracking processes of a streamlined airline. It incorporates structures, function-based modular programming, and user input handling.

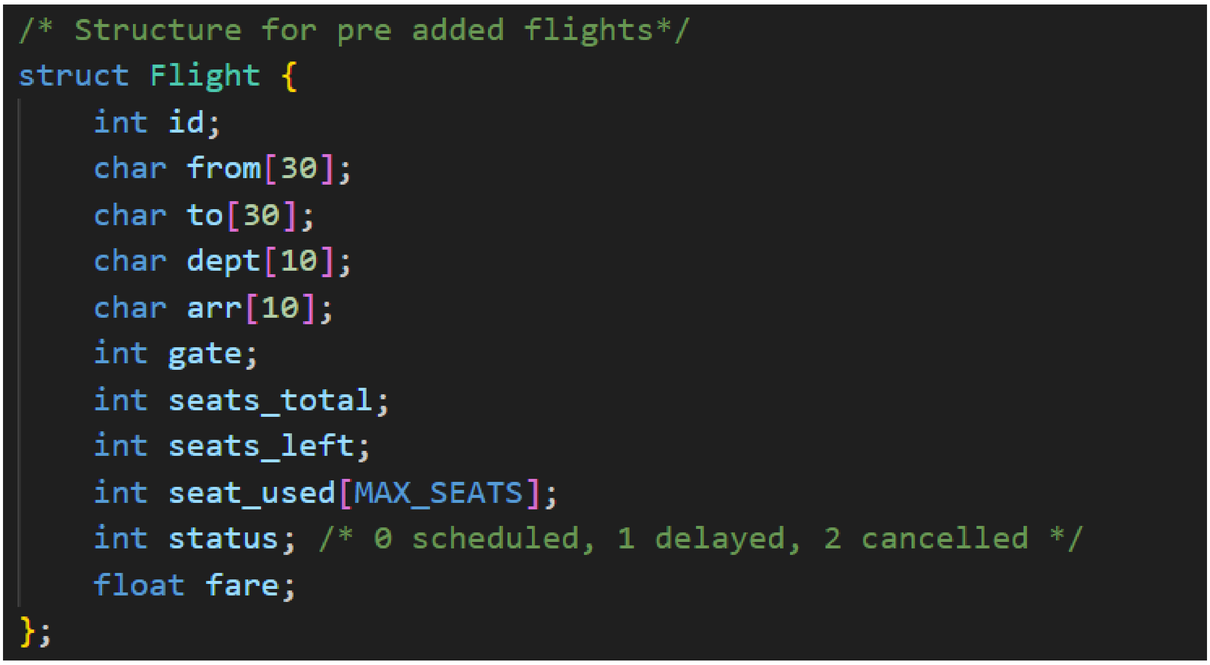
Upcoming Improvements

- Real-time flight seat availability
- Payment gateway simulation
- Baggage barcode generation
- Permanent file or database storage of records
- GUI version creation

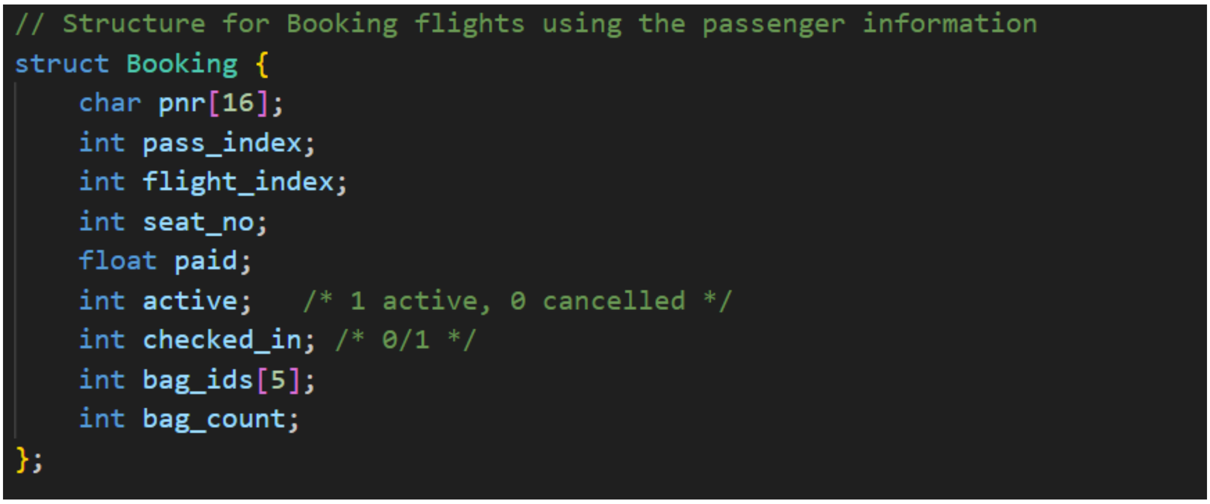
8. REFERENCES

- ANSI C Language Standards
- “Programming in C” by E. Balagurusamy
- Airline Check-In Workflow Documentation
- Online C Programming Resources
- Flowchart by eraser.c

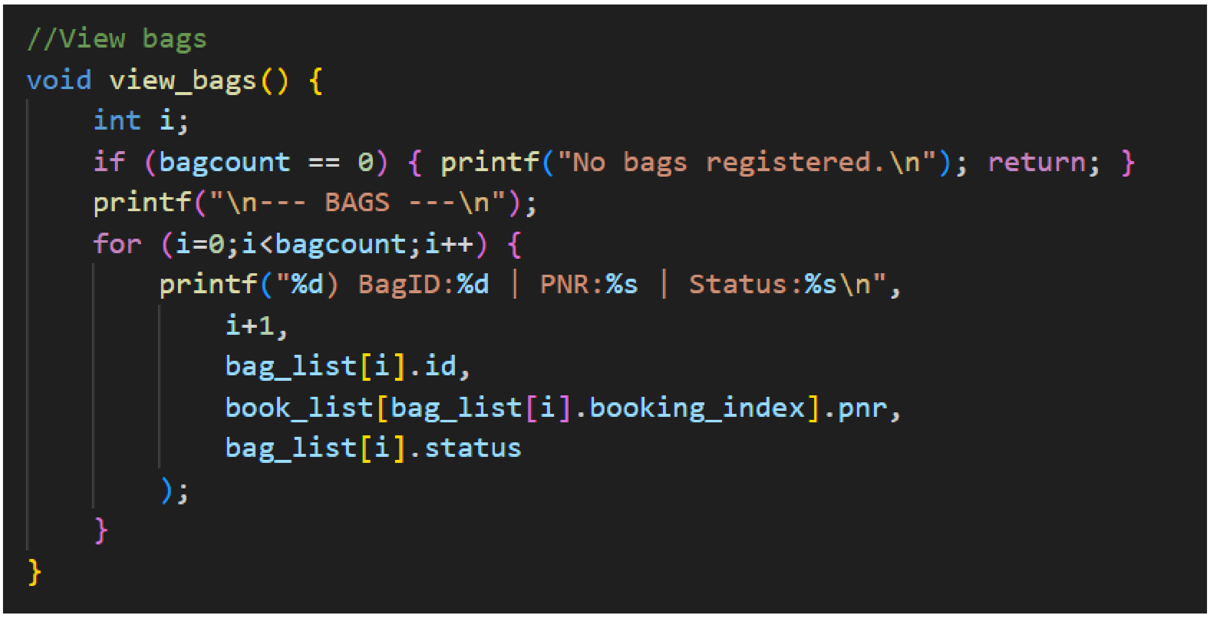
9. APPENDIX



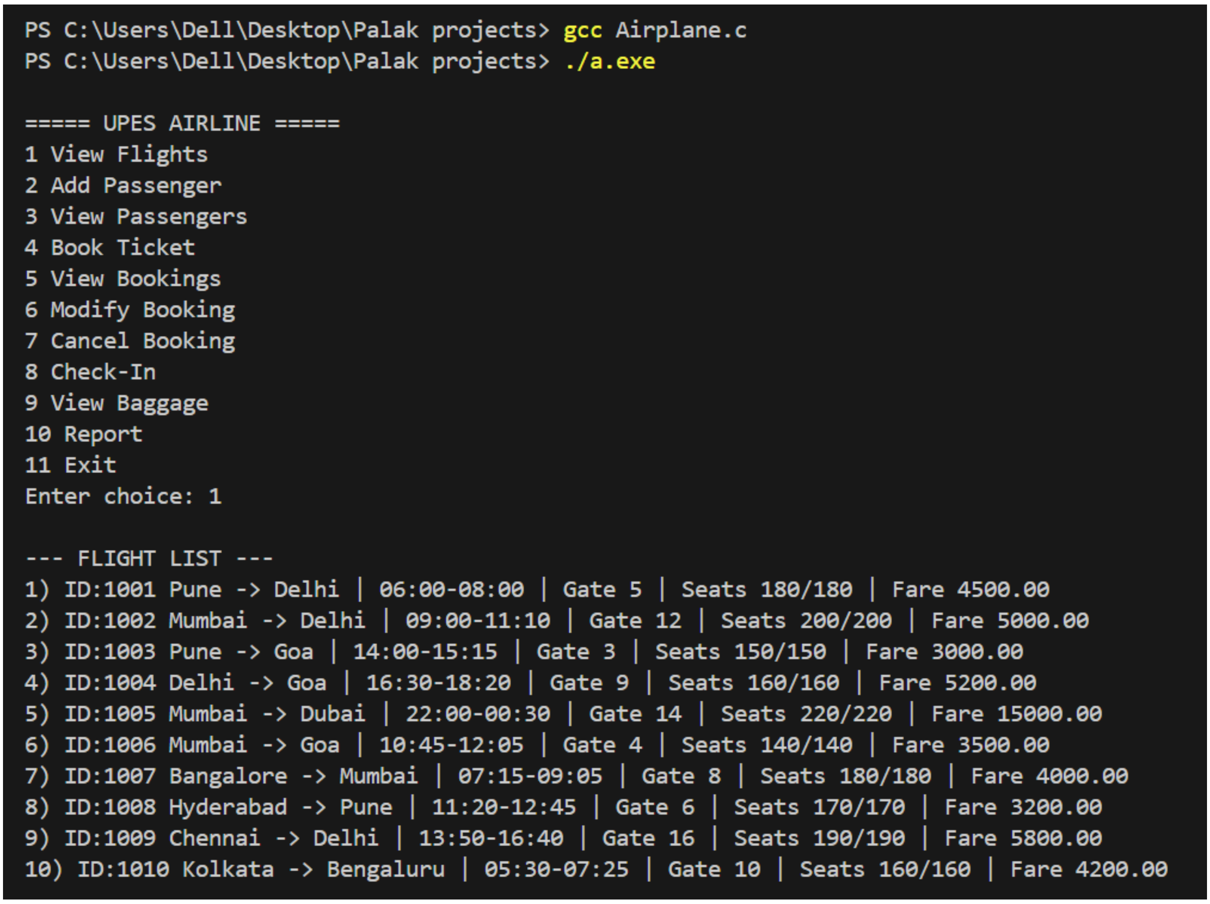
```c
/* Structure for pre added flights*/
struct Flight {
    int id;
    char from[30];
    char to[30];
    char dept[10];
    char arr[10];
    int gate;
    int seats_total;
    int seats_left;
    int seat_used[MAX_SEATS];
    int status; /* 0 scheduled, 1 delayed, 2 cancelled */
    float fare;
};
```



```c
// Structure for Booking flights using the passenger information
struct Booking {
    char pnr[16];
    int pass_index;
    int flight_index;
    int seat_no;
    float paid;
    int active;   /* 1 active, 0 cancelled */
    int checked_in; /* 0/1 */
    int bag_ids[5];
    int bag_count;
};
```



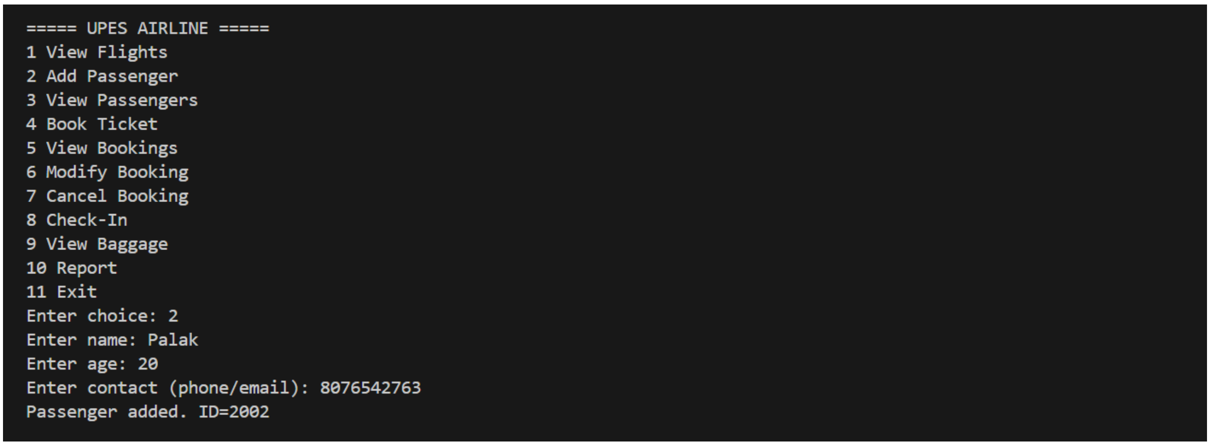
```c
//View bags
void view_bags() {
    int i;
    if (bagcount == 0) { printf("No bags registered.\n"); return; }
    printf("\n--- BAGS ---\n");
    for (i=0;i<bagcount;i++) {
        printf("%d) BagID:%d | PNR:%s | Status:%s\n",
            i+1,
            bag_list[i].id,
            book_list[bag_list[i].booking_index].pnr,
            bag_list[i].status
        );
    }
}
```



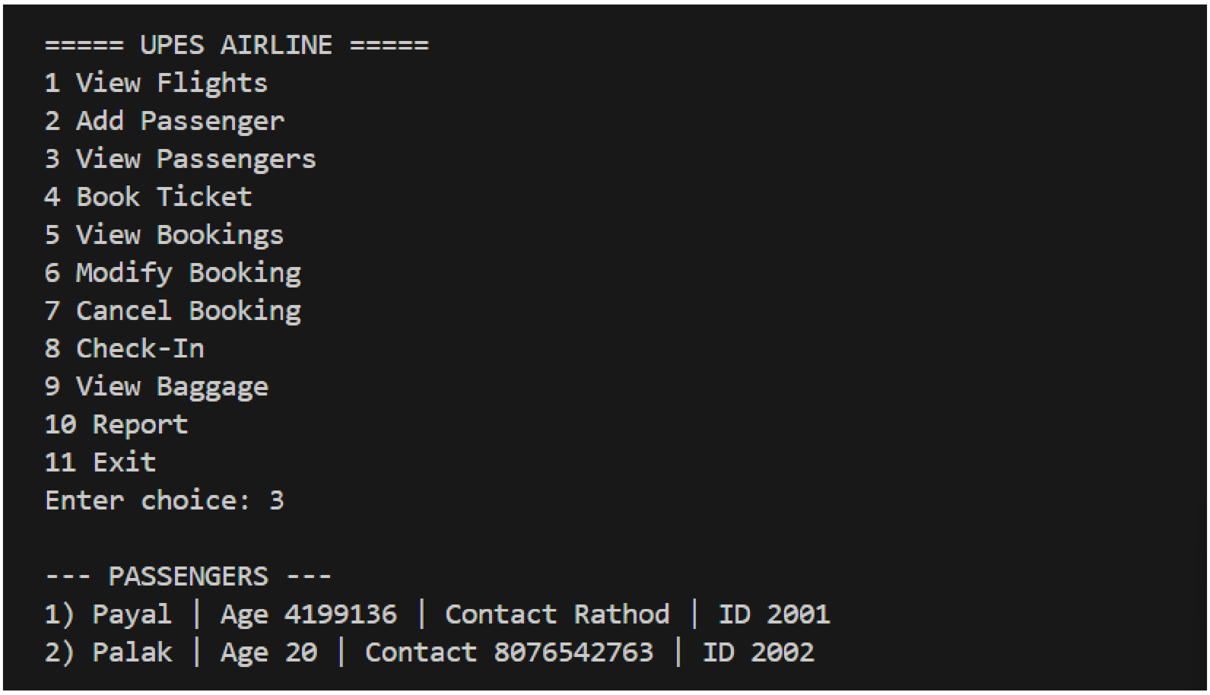
```
PS C:\Users\Dell\Desktop\Palak projects> gcc Airplane.c
PS C:\Users\Dell\Desktop\Palak projects> ./a.exe

===== UPES AIRLINE =====
1 View Flights
2 Add Passenger
3 View Passengers
4 Book Ticket
5 View Bookings
6 Modify Booking
7 Cancel Booking
8 Check-In
9 View Baggage
10 Report
11 Exit
Enter choice: 1

--- FLIGHT LIST ---
1) ID:1001 Pune -> Delhi | 06:00-08:00 | Gate 5 | Seats 180/180 | Fare 4500.00
2) ID:1002 Mumbai -> Delhi | 09:00-11:10 | Gate 12 | Seats 200/200 | Fare 5000.00
3) ID:1003 Pune -> Goa | 14:00-15:15 | Gate 3 | Seats 150/150 | Fare 3000.00
4) ID:1004 Delhi -> Goa | 16:30-18:20 | Gate 9 | Seats 160/160 | Fare 5200.00
5) ID:1005 Mumbai -> Dubai | 22:00-00:30 | Gate 14 | Seats 220/220 | Fare 15000.00
6) ID:1006 Mumbai -> Goa | 10:45-12:05 | Gate 4 | Seats 140/140 | Fare 3500.00
7) ID:1007 Bangalore -> Mumbai | 07:15-09:05 | Gate 8 | Seats 180/180 | Fare 4000.00
8) ID:1008 Hyderabad -> Pune | 11:20-12:45 | Gate 6 | Seats 170/170 | Fare 3200.00
9) ID:1009 Chennai -> Delhi | 13:50-16:40 | Gate 16 | Seats 190/190 | Fare 5800.00
10) ID:1010 Kolkata -> Bengaluru | 05:30-07:25 | Gate 10 | Seats 160/160 | Fare 4200.00
```

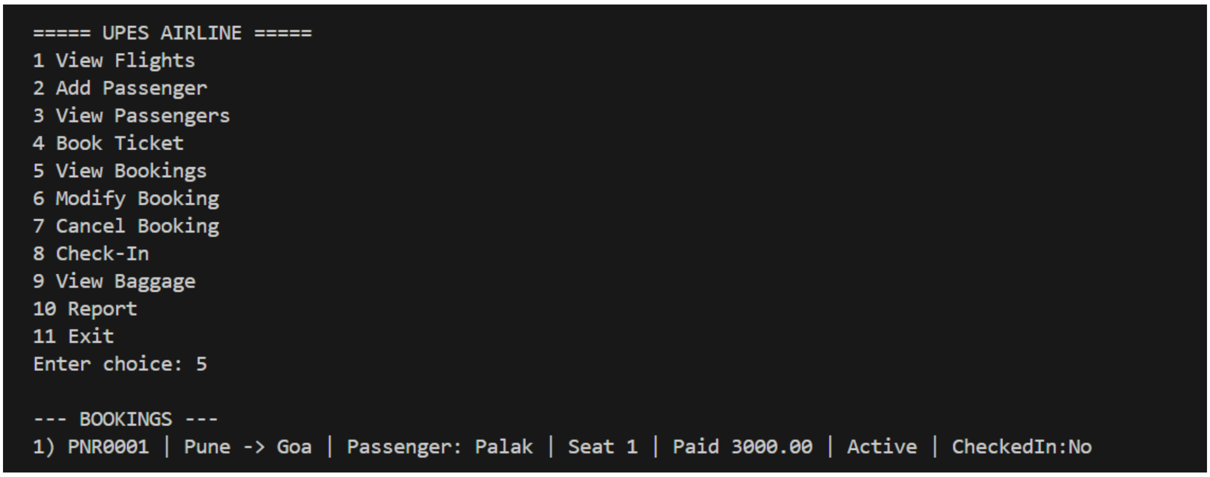


```
===== UPES AIRLINE =====
1 View Flights
2 Add Passenger
3 View Passengers
4 Book Ticket
5 View Bookings
6 Modify Booking
7 Cancel Booking
8 Check-In
9 View Baggage
10 Report
11 Exit
Enter choice: 2
Enter name: Palak
Enter age: 20
Enter contact (phone/email): 8076542763
Passenger added. ID=2002
```



```
===== UPES AIRLINE =====
1 View Flights
2 Add Passenger
3 View Passengers
4 Book Ticket
5 View Bookings
6 Modify Booking
7 Cancel Booking
8 Check-In
9 View Baggage
10 Report
11 Exit
Enter choice: 3

--- PASSENGERS ---
1) Payal | Age 4199136 | Contact Rathod | ID 2001
2) Palak | Age 20 | Contact 8076542763 | ID 2002
```



```
===== UPES AIRLINE =====
1 View Flights
2 Add Passenger
3 View Passengers
4 Book Ticket
5 View Bookings
6 Modify Booking
7 Cancel Booking
8 Check-In
9 View Baggage
10 Report
11 Exit
Enter choice: 5

--- BOOKINGS ---
1) PNR0001 | Pune -> Goa | Passenger: Palak | Seat 1 | Paid 3000.00 | Active | CheckedIn:No
```