Flight Ticket-Management System

Areeba, Rumaisa, Emman

Purpose

With millions of people boarding planes every day, air travel has become a necessity in the current epoch. It is crucial to verify the legitimacy and validity of airline tickets in order to uphold security, decrease fraud, and offer a flawless travel experience. We plan to create a dependable and effective C code application to fulfill this requirement by acting as a flight ticket validity checker, meant for utilisation by various airlines.

Display

- We commence with an intricately designed display

```
void printHeader()
{
    printf("*********************************************************************************\n");
    printf("*#################################################################################*\n");
    printf("*##*********************************************************************************##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                          FLIGHT TICKET SYSTEM                                 *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*                                                                               *##*\n");
    printf("*##*********************************************************************************##*\n");
    printf("*#################################################################################*\n");
    printf("*********************************************************************************\n");

}
```

Safety First!

- The user must enter a password to access the system

```c
//Check and compare passwords
int checkPassword() {
    char storedPassword[MAX_PASSWORD_LENGTH] = "Secure123";
    char enteredPassword[MAX_PASSWORD_LENGTH];

    // Prompt user for password
    printf("Enter your password: ");
    fgets(enteredPassword, MAX_PASSWORD_LENGTH, stdin);

    // Remove newline character from entered password
    enteredPassword[strcspn(enteredPassword, "\n")] = '\0';

    // Check password conditions
    int length = strlen(enteredPassword);
    int hasUppercase = 0;
    int hasLowercase = 0;
    int hasDigit = 0;

    for (int i = 0; i < length; ++i) {
        if (isupper(enteredPassword[i])) {
            hasUppercase = 1;
        } else if (islower(enteredPassword[i])) {
            hasLowercase = 1;
        } else if (isdigit(enteredPassword[i])) {
            hasDigit = 1;
        }
    }
```

Constituents

- **Various Validation Functions such as:**

```
//Seat and Row Validation
bool isValidSeat(char* ticket, int first_row, int last_row)
{
    char seat= getSeatNumber(ticket);
    int row= getRow(ticket);
    return (row >= first_row && row <= last_row) &&
    (seat==SA || seat==SB || seat==SC || seat==SD || seat==SE || seat==SF);
}
```

- **Various Data Acquiring Functions such as:**

```
//Extract row number from ticket as an integer
int getRow(char* ticket)
{
    int rowLen=2;
    int startIndex=14;
    char rowTemp[rowLen+1];
    //Copy the row characters to a temporary string
    memcpy(rowTemp, ticket + startIndex, rowLen);
    rowTemp[rowLen] = '\0'; //Null-terminate the temporary row string
    //Convert temporary string to integer using atoi function
    int row = atoi(rowTemp);
    return row;
}
```

- ## Function to Change Seat

```
//to change seat
void changeSeat(char* ticket, char* row_num, char seat)
{
    ticket[14]=row_num[0];
    ticket[15]=row_num[1];
    ticket[16]=seat;
}
```

- ## Function to change date

```
//to change date
char* changeDate(char* ticket, char* day, char* month, char* year)
{
    char* new_ticket= (char*)malloc(18 * sizeof(char));
    if (new_ticket==NULL)
    {
        fprintf(stderr, "Memory allocation failed\n");
        exit(EXIT_FAILURE);
    }
    sprintf(new_ticket, "%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c",
            year[0], year[1], year[2], year[3],
            month[0], month[1],
            day[0], day[1],
            ticket[8], ticket[9], ticket[10], ticket[11], ticket[12], ticket[13], ticket[14], ticket[15], ticket[16]);
    return new_ticket;
    free(new_ticket);
}
```

A Complete Overview… or SkyView?

- Our program extracts the following information from a ticket:
  - Date
  - Airport Departure Code
  - Airport Arrival Code
  - Row number
  - Seat Number
- It performs comprehensive validation checks on each of the above extracted pieces of information
- It allows seat and date changes
- NOTE: This program is specifically designed for an airline company!

Sample Output Using Test Case

```
Enter the number of the first row and the last row:
1 30
Date (YYYYMMDD) is: 20231121
Year (YYYY) is: 2023
Month (MM) is: 11
Day (DD) is: 21
Departure Aiport code is: ABC
Arrival Airport code is: DEF
Row number is: 25
Seat A
Seat type: Window
Is valid seat: true
Is valid date: true
Is valid ticket: true
Adjacent: false
Behind: true
Connecting flight: true
Press 's' to change seat, press any other alphabet to skip.
s
Enter new row number:
17
Enter new seat number:
D
New ticket number is: 20231121ABCDEF17D
Press 'd' to change date, press any other alphabet to skip.
d
Enter updated day:
22
Enter updated month:
11
Enter updated year:
2023
New ticket number is: 20231122ABCDEF17D
Program ended with exit code: 0
```

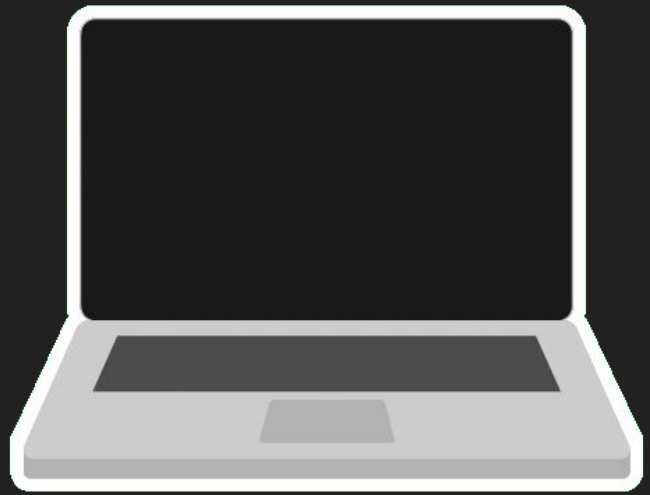
Reservation Functions

```
// Structure to store reservation details
struct FlightReservation {
    char passport[10];
    char name[30];
    char email[30];
    char destination[30];
    int seatNumber;
    struct FlightReservation* next;
};
```

-An effective and efficient system that aides the purpose of advanced bookings to help maintain traffic in the airbus.

Behind The Scenes

- We have already made use of the following C language tools:
- Functions
- Dynamic Memory Allocation
- Strings
- Pointers
- Loops
- Arrays
- Different data types



Before Departure…

Ponder on the significance of our code:

-Automation

-Human Error

-Data Preservation

-Compatibility

Safe travels !