

PROJECT: BUS RESERVATION SYSTEM

NAME: ABHAY GUPTA

SAP ID: 590026149

COURSE: B. TECH (CSE)

EMAIL ID: Abhay.26149@stu.upes.ac.in

SUBJECT: PROGRAMMING IN C

ABSTRACT:

Bus Reservation System is a tool that allows users to book tickets for their journey in advance. It offers multiple features to provide a hassle-free experience to a traveler. This article aims at building a rudimentary Bus Reservation System.

Components of the Bus Reservation System:

Login System: Users can access the system by entering their username and password. The program provides a collection of preconfigured users and their credentials.

Ticket Purchase: Logged-in individuals may reserve tickets for available buses by entering the bus number, their name, and age. The program allocates a seat number and decreases the number of available seats on the selected bus.

Cancel Tickets: By entering their name, users can cancel their tickets. The program expands the number of available seats while removing the passenger entry.

Checking Bus Status: Users may check the status of the bus they are currently scheduled to ride on. The program displays information such as the bus number, origin and destination, total number of seats, available seats, and fare.

The program employs frameworks to organize data for buses, passengers, and users. It also has menus for both the main menu (login) and the user menu (booking, cancelling, and checking status).

```

C:\main.c M>
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 // Define a structure to store bus information
6 struct Bus{
7     int busNumber;
8     char source[50];
9     char destination[50];
10    int totalSeats;
11    int availableSeats;
12    float fare;
13 };
14
15 // Define a structure to store user login information
16 struct User{
17     char username[50];
18     char password[50];
19 };
20
21 // Function to display the main menu
22 void displayMainMenu(){
23     printf("\n==== Main Menu ====\n");
24     printf("1. Login\n");
25     printf("2. Exit\n");
26     printf("Enter your choice: ");
27 }
28
29 // Function to display the user menu
30 void displayUserMenu(){
31     printf("\n==== User Menu ====\n");
32     printf("1. Book a Ticket\n");
33     printf("2. Cancel a Ticket\n");
34     printf("3. Check Bus Status\n");
35     printf("4. Logout\n");
36     printf("Enter your choice: ");
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

a.out + ▾

```

1. Login
2. Exit
Enter your choice: 1
Enter Username: user1
Enter Password: pass1
Login successful. Welcome, user1!
==== User Menu ====

```

PART-1 EXPLANATION:

Initialization of Structures and Variables:

- The code starts by defining three structures: struct Bus to store bus information, struct Passenger for passenger details, and struct User to store user login information.
- It also includes necessary libraries and declares functions for displaying menus, user login, booking tickets, cancelling tickets, and checking bus status.
- Three arrays are initialized:
 1. struct User users[5] stores user login credentials for five users.

2. struct Bus buses[3] stores information about three different buses.
3. struct Passenger passengers[500] is an array to store passenger details.
4. numUsers, numBuses, and numPassengers are variables to keep track of the number of users, buses, and passengers.
5. loggedInUserId is initially set to -1, indicating that no user is logged in.

```

C main.c M ●
C main.c > loginUser(User [], int, char [], char [])
30 void displayUserMenu(){
31
32     // Function to perform user login
33     int loginUser(struct User users[], int numUsers, char username[], char password[]){
34         for (int i = 0; i < numUsers; i++){
35             if (strcmp(users[i].username, username) == 0 && strcmp(users[i].password, password) == 0){
36                 return i; // Return the index of the logged-in user
37             }
38         }
39         return -1; // Login failed
40     }
41
42     // Function to book tickets
43     void bookTicket(struct Bus buses[], int numBuses){
44         printf("\nEnter Bus Number: ");
45         int busNumber;
46         scanf("%d", &busNumber);
47
48         // Find the bus with the given busNumber
49         int busIndex = -1;
50         for (int i = 0; i < numBuses; i++){
51             if (buses[i].busNumber == busNumber){
52                 busIndex = i;
53                 break;
54             }
55         }
56
57         if (busIndex == -1){
58             printf("Bus with Bus Number %d not found.\n", busNumber);
59         }
60         else{
61             printf("Enter Number of Seats: ");
62             int seatsToBook;
63             scanf("%d", &seatsToBook);
64
65             if (buses[busIndex].availableSeats < seatsToBook){
66                 printf("Not enough seats available for bus %d.\n", busNumber);
67             }
68             else{
69                 printf("Ticket booked successfully for bus %d, seat %d.\n", busNumber, seatsToBook);
70             }
71         }
72     }
73
74     PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
75     === User Menu ===
76     1. Book a Ticket
77     2. Cancel a Ticket
78     3. Check Bus Status
79     4. Logout
80     Enter your choice: 1
81     Enter Bus Number: 101
82     Enter Number of Seats: 2
83
84     a.out + ~ [ ]

```

PART-2 EXPLANATION:

Main Function and Program Loop:

- The ‘main’ function is the entry point of the program.

- It initializes the user data, bus data, passenger data, and the `loggedInUserId` variable.
- The program enters a `while (1)` loop, which serves as the main program loop and runs until the user chooses to exit.

```

C main.c M
C main.c > ⌂ loginUser(User [], int, char [], char [])
50 void bookTicket(struct Bus buses[], int numBuses){
51     else{
52         if (buses[busIndex].availableSeats < seatsToBook){
53             printf("Sorry, only %d seats are available.\n", buses[busIndex].availableSeats);
54         }
55         else{
56             buses[busIndex].availableSeats -= seatsToBook;
57             printf("Booking successful! %d seats booked on Bus Number %d.\n", seatsToBook, busNumber);
58         }
59     }
60 }
61
62 // Function to cancel tickets
63 void cancelTicket(struct Bus buses[], int numBuses){
64     printf("\nEnter Bus Number: ");
65     int busNumber;
66     scanf("%d", &busNumber);
67
68     // Find the bus with the given busNumber
69     int busIndex = -1;
70     for (int i = 0; i < numBuses; i++){
71         if (buses[i].busNumber == busNumber){
72             busIndex = i;
73             break;
74         }
75     }
76
77     if (busIndex == -1){
78         printf("Bus with Bus Number %d not found.\n", busNumber);
79     }
80     else{
81         printf("Enter Number of Seats to Cancel: ");
82         int seatsToDelete;
83
84         if (buses[busIndex].availableSeats < seatsToDelete){
85             printf("Not enough seats available on Bus Number %d.\n", busNumber);
86         }
87         else{
88             buses[busIndex].availableSeats -= seatsToDelete;
89             printf("Cancellation successful! %d seats canceled on Bus Number %d.\n", seatsToDelete, busNumber);
90         }
91     }
92 }
93
94
95
96
97
98
99
100
101
102

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS a.out + v

```

== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 2

Enter Bus Number: 101
Enter Number of Seats to Cancel: 2
Cancellation successful! 2 seats canceled on Bus Number 101.

```

PART-3 EXPLANATION:

Main Menu (displayMainMenu):

- If no user is logged in (`loggedInUserId == -1`), the program displays the main menu with options to "Login" or "Exit."

```

C main.c M ●
C main.c > ⌂ loginUser(User [], int, char [], char [])
83 void cancelTicket(struct Bus buses[], int numBuses){
100 else{
102     int seatsToCancel;
103     scanf("%d", &seatsToCancel);
104
105     if (seatsToCancel > (buses[busIndex].totalSeats - buses[busIndex].availableSeats)){
106         printf("Error: You can't cancel more seats than were booked.\n");
107     }
108     else{
109         buses[busIndex].availableSeats += seatsToCancel;
110         printf("Cancellation successful! %d seats canceled on Bus Number %d.\n", seatsToCancel,
111             busNumber);
112     }
113 }
114 }

// Function to check bus status
void checkBusStatus(struct Bus buses[], int numBuses){
    printf("\nEnter Bus Number: ");
    int busNumber;
    scanf("%d", &busNumber);

    // Find the bus with the given busNumber
    int busIndex = -1;
    for (int i = 0; i < numBuses; i++){
        if (buses[i].busNumber == busNumber){
            busIndex = i;
            break;
        }
    }
    if (busIndex != -1){
        printf("\nBus Number: %d\n", buses[busIndex].busNumber);
    }
}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
a.out + v

== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 4
Logging out.

== Main Menu ==
1. Login
2. Exit

```

PART-4 EXPLANATION:

User Login (loginUser):

- If the user selects "Login" (choosing option 1), the program prompts the user to enter their username and password.
- The `loginUser` function is called, which checks if the provided username and password match any of the predefined users in the `users` array.
- If a match is found, the `loggedInUserId` is set to the index of the logged-in user, and a welcome message is displayed.
- If no match is found, a login failed message is shown, and the user can try again or choose to exit.

```

C main.c M •
C main.c > ⚡ loginUser(User [], int, char [], char [])
117 void checkBusStatus(struct Bus buses[], int numBuses){
131     if (busIndex != -1){
133         printf("Source: %s\n", buses[busIndex].source);
134         printf("Destination: %s\n", buses[busIndex].destination);
135         printf("Total Seats: %d\n", buses[busIndex].totalSeats);
136         printf("Available Seats: %d\n", buses[busIndex].availableSeats);
137         printf("Fare: %.2f\n", buses[busIndex].fare);
138     }
139     else{
140         printf("Bus with Bus Number %d not found.\n", busNumber);
141     }
143 }
144 int main(){
145     // Initialize user data
146     struct User users[5] = {
147         {"user1", "pass1"}, {"user2", "pass2"}, {"user3", "pass3"}, {"user4", "pass4"}, {"user5", "pass5"}
148     };
149     int numUsers = 5;
150
151     // Initialize bus data
152     struct Bus buses[3] = {
153         {101, "City A", "City B", 50, 50, 500.0},
154         {102, "City C", "City D", 40, 40, 400.0},
155         {103, "City E", "City F", 30, 30, 300.0},
156     };
157     int numBuses = 3;
158 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS a.out + □
```

```

== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 4
Logging out.

== Main Menu ==
1. Login

```

PART-5 EXPLANATION:

User Menu (displayUserMenu):

- If a user is logged in (`loggedInUserId >= 0`), the program displays the user menu with options to "Book a Ticket," "Cancel a Ticket," "Check Bus Status," or "Logout."

```

C main.c M ●
C main.c > ⌂ loginUser(User [], int, char [], char [])
144 int main(){
159     int loggedInUserId = -1; // Index of the logged-in user
160
161     while (1){
162         if (loggedInUserId == -1){
163             displayMainMenu();
164             int choice;
165             scanf("%d", &choice);
166
167             if (choice == 1){
168                 char username[50];
169                 char password[50];
170
171                 printf("Enter Username: ");
172                 scanf("%s", username);
173                 printf("Enter Password: ");
174                 scanf("%s", password);
175
176                 loggedInUserId = loginUser(users, numUsers, username, password);
177                 if (loggedInUserId == -1){
178                     printf("Login failed. Please check your username and password.\n");
179                 }
180                 else{
181                     printf("Login successful. Welcome, %s!\n", username);
182                 }
183             }
184             else if (choice == 2){
185                 printf("Exiting the program.\n");
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
a.out + □

```

Enter your choice: 3
Enter Bus Number: 101
Bus Number: 101
Source: City A
Destination: City B
Total Seats: 50
Available Seats: 50
Fare: 500.00

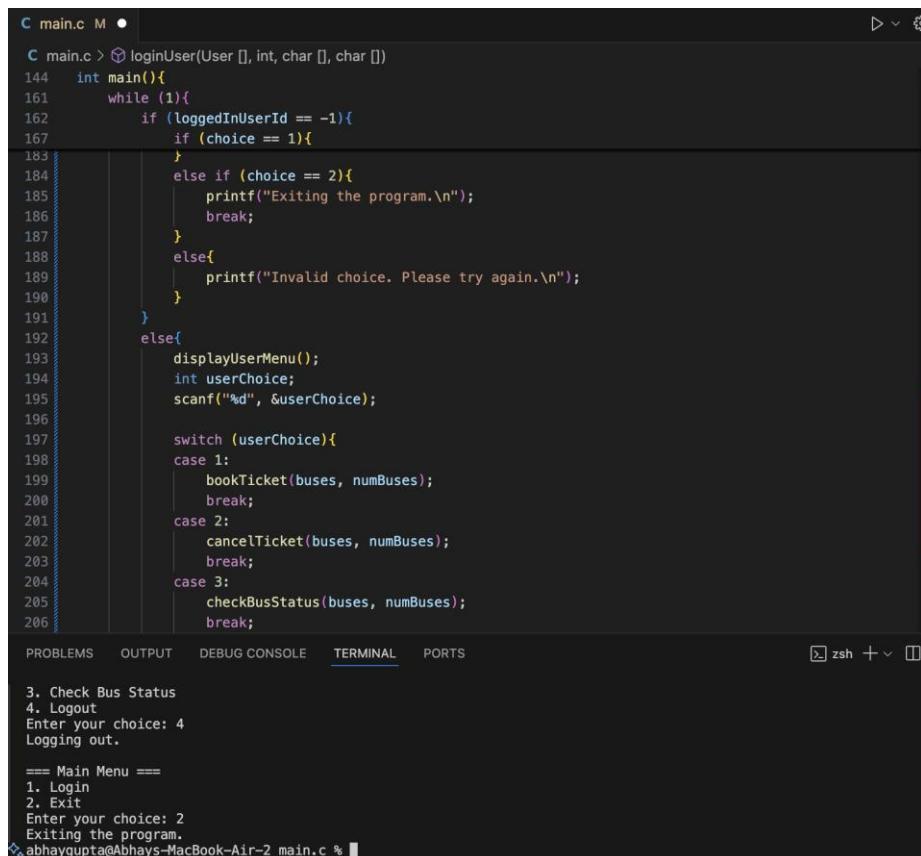
PART-6 EXPLANATION:

Booking a Ticket (bookTicket):

- If the user selects "Book a Ticket" (option 1 from the user menu), the program asks for the bus number they want to book a ticket for.
- It then searches for the selected bus in the `buses` array and checks if there are available seats.
- If there are available seats, the program prompts the user to enter their name and age. It assigns a seat number, records

the passenger's bus number, and updates the available seats.

- A success message is displayed, and the number of passengers (`numPassengers`) is incremented.



```
C main.c M ●
C main.c > loginUser(User [], int, char [], char [])
144 int main(){
145     while (1){
146         if (loggedInUserId == -1){
147             if (choice == 1){
148                 }
149                 else if (choice == 2){
150                     printf("Exiting the program.\n");
151                     break;
152                 }
153                 else{
154                     printf("Invalid choice. Please try again.\n");
155                 }
156             }
157             else{
158                 displayUserMenu();
159                 int userChoice;
160                 scanf("%d", &userChoice);
161
162                 switch (userChoice){
163                     case 1:
164                         bookTicket(buses, numBuses);
165                         break;
166                     case 2:
167                         cancelTicket(buses, numBuses);
168                         break;
169                     case 3:
170                         checkBusStatus(buses, numBuses);
171                         break;
172                 }
173             }
174         }
175     }
176 }
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS zsh + □

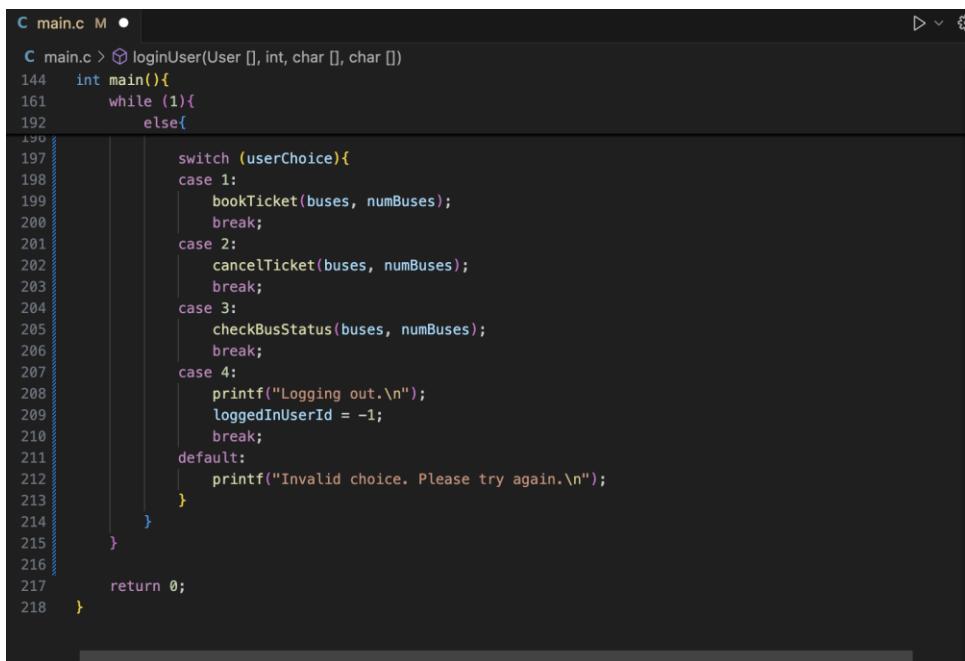
```
3. Check Bus Status
4. Logout
Enter your choice: 4
Logging out.

== Main Menu ==
1. Login
2. Exit
Enter your choice: 2
Exiting the program.
abhaygupta@Abhays-MacBook-Air-2:~/Desktop/main$
```

PART-7 EXPLANATION:

Canceling a Ticket (cancelTicket):

- If the user selects "Cancel a Ticket" (option 2 from the user menu), the program asks for the passenger's name.
- It then searches for the passenger in the `passengers` array, ensuring that the passenger is on the bus of the currently logged-in user. If found, the ticket is canceled, the available seats are increased, and the passenger entry is removed.



```

C main.c M •
C main.c > ⌂ loginUser(User [], int, char [], char [])
144 int main(){
161     while (1){
192         else{
193             switch (userChoice){
194                 case 1:
195                     bookTicket(buses, numBuses);
196                     break;
197                 case 2:
198                     cancelTicket(buses, numBuses);
199                     break;
200                 case 3:
201                     checkBusStatus(buses, numBuses);
202                     break;
203                 case 4:
204                     printf("Logging out.\n");
205                     loggedInUserId = -1;
206                     break;
207                 default:
208                     printf("Invalid choice. Please try again.\n");
209             }
210         }
211     }
212 }
213
214 return 0;
215
216
217
218 }

```

PART-8 EXPLANATION:

Checking Bus Status (checkBusStatus):

- If the user selects "Check Bus Status" (option 3 from the user menu), the program asks the user to enter the bus

number and then displays information about the bus number the user entered. This includes the bus number, source, destination, total seats, available seats, and fare.

Logging Out:

- If the user selects "Logout" (option 4 from the user menu), they are logged out, and `loggedInUserId` is set back to -1.

Exiting the Program:

- If the user selects "Exit" from the main menu (option 2), the program exits the `while (1)` loop and ends.

Test Cases

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

● abhaygupta@Abhays-MacBook-Air-2 main.c % gcc main.c
◇ abhaygupta@Abhays-MacBook-Air-2 main.c % ./a.out

    === Main Menu ===
1. Login
2. Exit
Enter your choice: 1
Enter Username: user1
Enter Password: pass1
Login successful. Welcome, user1!

    === User Menu ===
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 1

Enter Bus Number: 101
Enter Number of Seats: 2
Booking successful! 2 seats booked on Bus Number 101.
```

```
== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 2

Enter Bus Number: 101
Enter Number of Seats to Cancel: 2
Cancellation successful! 2 seats canceled on Bus Number 101.
```

```
== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 3

Enter Bus Number: 101

Bus Number: 101
Source: City A
Destination: City B
Total Seats: 50
Available Seats: 50
Fare: 500.00
```

```
== User Menu ==
1. Book a Ticket
2. Cancel a Ticket
3. Check Bus Status
4. Logout
Enter your choice: 4
Logging out.

== Main Menu ==
1. Login
2. Exit
Enter your choice: 2
Exiting the program.
abhaygupta@Abhays-MacBook-Air-2 main.c %
```