1)
```c
#include <stdio.h>
#include <string.h>
#define MAX_SEATS 10
struct Passenger {
    char name[50];
    int age;
    char contact[15];
};
struct Bus {
    int busNumber;
    char route[50];
    char departureTime[10];
    int seats[MAX_SEATS];  // 0 means available, 1 means booked
    struct Passenger passengers[MAX_SEATS];
};
void bookSeat(struct Bus *bus) {
    int seatNumber;
    printf("Enter seat number (1-%d): ", MAX_SEATS);
    scanf("%d", &seatNumber);
    if (seatNumber < 1 || seatNumber > MAX_SEATS || (*bus).seats[seatNumber - 1] == 1) {
        printf("Invalid or already booked!\n");
        return;}
    printf("Enter Name: ");
    scanf("%s", (*bus).passengers[seatNumber - 1].name);
    printf("Enter Age: ");
    scanf("%d", &(*bus).passengers[seatNumber - 1].age);
    printf("Enter Contact Number: ");
    scanf("%s", (*bus).passengers[seatNumber - 1].contact);
    (*bus).seats[seatNumber - 1] = 1;  // Mark seat as booked
    printf("Seat %d booked successfully!\n", seatNumber);
}
void cancelSeat(struct Bus *bus) {
```

```c
    int seatNumber;
    printf("Enter seat number to cancel (1-%d): ", MAX_SEATS);
    scanf("%d", &seatNumber);
    if (seatNumber < 1 || seatNumber > MAX_SEATS || (*bus).seats[seatNumber - 1] == 0) {
        printf("Invalid or not booked!\n");
        return;
    }
    (*bus).seats[seatNumber - 1] = 0;  // Mark seat as available
    printf("Booking for seat %d canceled.\n", seatNumber);
}
void displaySeats(struct Bus *bus) {
    printf("\nBooked Seats:\n");
    for (int i = 0; i < MAX_SEATS; i++) {
        if ((*bus).seats[i] == 1) {
            printf("Seat %d: %s, Age: %d, Contact: %s\n",
                i + 1, (*bus).passengers[i].name, (*bus).passengers[i].age, (*bus).passengers[i].contact);
        }
    }
}
int main() {
    struct Bus myBus = {101, "City A to City B", "10:00 AM", {0}};
    int choice;
    do {
        printf("\n1. Book Seat\n2. Cancel Booking\n3. View Bookings\n4. Exit\nEnter choice: ");
        scanf("%d", &choice);
        switch (choice) {
            case 1: bookSeat(&myBus); break;
            case 2: cancelSeat(&myBus); break;
            case 3: displaySeats(&myBus); break;
            case 4: printf("Exiting...\n"); break;
            default: printf("Invalid choice! Try again.\n");
        }
    }
```

```c
    } while (choice != 4);
    return 0;
}


2)  #include <stdio.h>
#include <limits.h>
#define MAX 10
void dijkstra(int graph[MAX][MAX], int src, int dest, int n) {
    int dist[MAX], visited[MAX], prev[MAX], u, v;
    for (v = 0; v < n; v++) dist[v] = INT_MAX, visited[v] = 0, prev[v] = -1;
    dist[src] = 0;
    for (int i = 0; i < n - 1; i++) {
        u = -1;
        for (v = 0; v < n; v++)
            if (!visited[v] && (u == -1 || dist[v] < dist[u])) u = v;
        if (u == -1) break;
        visited[u] = 1;
        for (v = 0; v < n; v++)
            if (graph[u][v] && dist[u] + graph[u][v] < dist[v]) {
                dist[v] = dist[u] + graph[u][v];
                prev[v] = u;
            }
    }
    printf("\nRoute: ");
    for (v = dest; v != -1; v = prev[v]) printf("%d ", v);
    printf("\nTime: %d mins\n", dist[dest]);
}
int main() {
    int n, graph[MAX][MAX], src, dest;
    printf("Stops: ");
    scanf("%d", &n);
    printf("Distances:\n");
```

```c
    for (int i = 0; i < n; i++)

        for (int j = 0; j < n; j++)

            scanf("%d", &graph[i][j]);

    printf("Source Dest: ");

    scanf("%d %d", &src, &dest);

    dijkstra(graph, src, dest, n);

    return 0;

}
```

3)
```c
#include <stdio.h>

struct Ticket {

    char name[50];

    int age;

    char contact[15];

    float price;

};

int main() {

    struct Ticket ticket;

    int choice;

    printf("Enter Passenger Name: ");

    scanf("%s", ticket.name);

    printf("Enter Age: ");

    scanf("%d", &ticket.age);

    printf("Enter Contact Number: ");

    scanf("%s", ticket.contact);

    printf("Enter Ticket Price: ");

    scanf("%f", &ticket.price);

    printf("\nTicket Booked!\n");

    printf("Passenger Name: %s\n", ticket.name);

    printf("Ticket Price: %.2f\n", ticket.price);

    printf("Enter 1 to Pay or 2 to Cancel: ");

    scanf("%d", &choice);

    if (choice == 1) {
```

```c
        printf("Payment Successful!\n");
    } else {
        printf("Ticket Cancelled.\n");
    }
    return 0;
}
```

4)
```c
#include <stdio.h>
typedef struct {
    int busID;
    double latitude;
    double longitude;
    char timestamp[20];
} Bus;
void updateBusLocation(Bus *bus, double lat, double lon, const char *timestamp) {
    bus->latitude = lat;
    bus->longitude = lon;
    snprintf(bus->timestamp, sizeof(bus->timestamp), "%s", timestamp);
}
void displayBusInfo(Bus bus) {
    printf("Bus ID: %d\n", bus.busID);
    printf("Current Location: Latitude %.6f, Longitude %.6f\n", bus.latitude, bus.longitude);
    printf("Last Updated: %s\n", bus.timestamp);
}
int main() {
    Bus bus1;
    bus1.busID = 101;
    printf("Enter bus location (latitude longitude): ");
    scanf("%lf %lf", &bus1.latitude, &bus1.longitude);
    printf("Enter timestamp (YYYY-MM-DD HH:MM): ");
    scanf("%s", bus1.timestamp);
    displayBusInfo(bus1);
    double newLat, newLon;
```

```c
    char newTimestamp[20];
    printf("\nEnter updated bus location (latitude longitude): ");
    scanf("%lf %lf", &newLat, &newLon);
    printf("Enter updated timestamp (YYYY-MM-DD HH:MM): ");
    scanf("%s", newTimestamp);
    updateBusLocation(&bus1, newLat, newLon, newTimestamp);
    displayBusInfo(bus1);
    return 0;
}
```