Scenario 1

```
#include<stdio.h>
#include<stdlib.h>
int n,totalseat=10;
struct bus{
   int busNum,numofseats[20],age;
   char root[100],name[100],departureTime[100];
   unsigned long int phno;
}s;
void bussDetails(){
   printf("Enter the bus number\n");
   scanf("%d",&s.busNum);
   printf("Enter the root\n");
   scanf("%s",&s.root);
   printf("Enter the departure time\n");
   scanf("%s",s.departureTime);
   printf("Enter the no of seats\n");
   scanf("%d",&n);
   for(int i=0;i<n;i++)
   {
        int seatno;
        printf("Enter seat number %d: ", i + 1);
        scanf("%d",&seatno);
        if (s.numofseats[seatno ] == 1) {
          printf("Seat %d is already taken.Please choose another seat.\n", seatno);
        }
        else{
          s.numofseats[seatno]=1;
        }

   }
   if (totalseat> 10) {
      printf("Only 20 seats available. Setting to 20.\n");
      totalseat= 10;
   }
```

```
}
void passengerDetails(){
   printf("Enter your name\n");
   scanf("%s",s.name);
   printf("Enter your age\n");
   scanf("%d",&s.age);
   printf("Enter your phno\n");
   scanf("%lu",&s.phno);
}
void display(){
   printf("buss details\nbusnum=%d\nroot=%s\ndeparture
time=%s\n",s.busNum,s.root,s.departureTime);
   printf("Seats: ");
   for (int i = 0; i < totalseat; i++) {
      printf("%d ", s.numofseats[i]);
   }
   printf("\npassenger
deatils\nname=%s\nage=%d\nphno=%lu\n",s.name,s.age,s.phno);
}
void cancellation(){
   int cancel;
   printf("Enter the seat num to cancel\n");
   scanf("%d",&cancel);
   if (s.numofseats[cancel] == 0)
   {printf("Seat %d is already vacant.\n", cancel); }
   else{
      s.numofseats[cancel]=0;
      printf("Seat cancelled successfully\n");
 }
}
void main(){
   int choice;
   while(1){
   printf("\n\nEnter the choice\n1)buss details\n2)passenger
details\n3)display\n4)seat cancellation\n5)to go out\n");
   scanf("%d",&choice);
   if(choice==1)
```

```
    { bussDetails();}
    if(choice==2)
    {passengerDetails();}
    if(choice==3)
    {display();}
    if(choice==4)
    { cancellation();}
    if(choice==5)
    { exit(0);}
    }
}
```

Scenario 3

```
#include <stdio.h>
#include <string.h>
#define MAX_TICKETS 100
typedef struct {
 char name[20];
 int age;
 char contact[10];
 char travelDate[9];
 char busType[20];
 float price;
} Ticket;
typedef struct {
 char paymentMethod[20];
 float amount;
 char transactionID[15];
} Payment;
Ticket tickets[MAX_TICKETS];
int ticketCount = 0;
float calculatePrice(const char *busType) {
if (strcmp(busType, "Luxury") == 0)
 return 150.0;
 else if (strcmp(busType, "Semi-Luxury") == 0)
 return 100.0;
```

```
 else
 return 50.0;
}
void bookTicket() {
 if (ticketCount >= MAX_TICKETS) {
 printf("No more tickets available!\n");
 return;
 }
 Ticket *t = &tickets[ticketCount];
 printf("Enter name, age, contact, travel date, and bus type
(Luxury/Semi-Luxury/Regular): ");
 scanf("%s %d %s %s %s", t->name, &t->age, t->contact, t->travelDate,
t->busType);
 t->price = calculatePrice(t->busType);
 printf("Ticket booked! Price: $%.2f\n", t->price);
 ticketCount++;
}
void processPayment() {
 if (ticketCount == 0) {
 printf("No tickets booked yet!\n");
 return;
 }
 Payment p;
 printf("Enter payment method, amount, and transaction ID: ");
 scanf("%s %f %s", p.paymentMethod, &p.amount, p.transactionID);

 printf("Payment successful!\nTransaction ID: %s\nAmount Paid: $%.2f\n",
p.transactionID,
p.amount);
}
void cancelTicket() {
 if (ticketCount == 0) {
 printf("No tickets to cancel!\n");
 return;
 }

 int ticketNumber;
```

```
 printf("Enter ticket number to cancel (1 to %d): ", ticketCount);
 scanf("%d", &ticketNumber);

 if (ticketNumber < 1 || ticketNumber > ticketCount) {
 printf("Invalid ticket number!\n");
 return;
 }

 for (int i = ticketNumber - 1; i < ticketCount - 1; i++) {
 tickets[i] = tickets[i + 1];
 }
 ticketCount--;
 printf("Ticket %d canceled. Refund processed.\n", ticketNumber);
}
int main() {
 int choice;
while (choice != 4) {
 printf("\n1. Book Ticket\n2. Process Payment\n3. Cancel Ticket\n4.
Exit\nChoice: ");
 scanf("%d", &choice);

 switch (choice) {
 case 1: bookTicket(); break;
 case 2: processPayment(); break;
 case 3: cancelTicket(); break;
 case 4: printf("Exiting...\n"); break;
 default: printf("Invalid choice!\n");
 }
 }

 return 0;
}
```