MINI PROJECT ON HOTEL MANAGEMENT

#include<stdio.h>

#include<string.h>

struct guest{

   char name[30];

   char mobile\_no[10];

   char purpose[10];

   int room\_no;

   int days;

   int bill;

};

struct room{

   int room\_no;

   int price;

   int add\_price;

   int status; // 0 for available; 1 for occupied.

};

struct amount{

   char name[10];

   int price;

   int count;

};

int room\_initialize(){

   FILE \*f=fopen("room.bin","wb");

   struct room ac[3],non\_ac[3];

   for(int i=0; i<3; i++){

    ac[i].room\_no=i+1;

    ac[i].price=1500;

    ac[i].add\_price=500;

    ac[i].status=0;

    non\_ac[i].room\_no=i+4;

    non\_ac[i].price=1400;

    non\_ac[i].add\_price=0;

    non\_ac[i].status=0;

   }

   struct room store={0,0,0,0};

   fwrite(&store,sizeof(store),1,f);

   for(int i=0; i<3; i++){

    fwrite(&ac[i],sizeof(ac[i]),1,f);

   }

   for(int i=0; i<3; i++){

    fwrite(&non\_ac[i],sizeof(ac[i]),1,f);

   }

   fclose(f);

   return 0;

}

int check\_in(){

struct room assign;

FILE \*f=fopen("room.bin","rb");

if(f==NULL){

puts("Something went wrong!");

return 0;

}

int ac=0,non\_ac=0;

fseek(f,sizeof(assign),SEEK\_SET);

for(int i=0; i<3; i++){

fread(&assign,sizeof(assign),1,f);

if(assign.status==0)

ac++;

}

for(int i=0; i<3; i++){

fread(&assign,sizeof(assign),1,f);

if(assign.status==0)

non\_ac++;

}

fclose(f);

printf("\nSelect a Room:\n1) AC - Available %d\n2) Non-AC - Available %d\nEnter 0) Cancel booking\n",ac,non\_ac);

int choice=1;

while(choice){

scanf("%d",&choice);

if(choice==1)

if(ac==0)

printf("Sorry! All the AC rooms are already accomodated\nEnter the choice again!\n");

else{

puts("Checking in AC room!");

f=fopen("room.bin","r+");

fseek(f,sizeof(assign),SEEK\_SET);

for(int i=1; i<4; i++){

fread(&assign,sizeof(assign),1,f);

if(assign.status==0){

assign.status=1;

fseek(f,-1\*sizeof(assign),SEEK\_CUR);

fwrite(&assign,sizeof(assign),1,f);

printf("Successfully checked in room number %d",i);

fclose(f);

return(i);

}

}

}

if(choice==2)

if(non\_ac==0)

printf("Sorry! All the Non-AC rooms are already accomodated\nEnter the choice again!\n");

else{

puts("Checking in Non-AC room!");

f=fopen("room.bin","r+");

fseek(f,sizeof(assign)\*4,SEEK\_CUR);

for(int i=4; i<7; i++){

fread(&assign,sizeof(assign),1,f);

if(assign.status==0){

assign.status=1;

fseek(f,-1\*sizeof(assign),SEEK\_CUR);

fwrite(&assign,sizeof(assign),1,f);

printf("Successfully checked in room number %d",i);

fclose(f);

return(i);

}

}

}

}

return 0;

}

void food\_initialize(){

   struct amount tea,coffee,b1,b2,l1,l2,d1,d2;

   char t[3]="Tea",c[6]="Coffee",br1[15]="Veg Breakfast1",br2[15]="Veg Breakfast2";

   char lu1[9]="Veg Lunch",lu2[8]="NV Lunch",di1[10]="Veg Dinner",di2[9]="NV Dinner";

   strcpy(tea.name,t);

   tea.price=15;

   strcpy(coffee.name,c);

   coffee.price=10;

   strcpy(b1.name,br1);

   b1.price=60;

   strcpy(b2.name,br2);

   b2.price=80;

   strcpy(l1.name,lu1);

   l1.price=50;

   strcpy(l2.name,lu2);

   l2.price=70;

   strcpy(d1.name,di1);

   d1.price=100;

   strcpy(d2.name,di2);

   d2.price=120;

}

void customer\_initialize(){

   FILE \*fp;

   fp=fopen("register.bin","wb");

   struct guest dummy={"dummy","9440593940","garbage",0,0,0};

   fwrite(&dummy,sizeof(dummy),1,fp);

   fclose(fp);

}

void customer\_registration(){

   FILE \*fp;

   char name[20],mobile[10],purpose[10];

   fp=fopen("register.bin","ab");

   struct guest value;

   printf("\n Enter your name: ");

   scanf("%s",name);

   strcpy(value.name,name);

   printf("\n Enter your mobile number: ");

   scanf("%s",mobile);

   strcpy(value.mobile\_no,mobile);

   printf("\n Enter your purpose of stay: ");

   scanf("%s",purpose);

   strcpy(value.purpose,purpose);

   printf("\n Enter the number of days of stay: ");

   scanf("%d",&value.days);

value.room\_no=check\_in();

if(value.room\_no<4) value.bill=value.days\*2000;

else value.bill=value.days\*1400;

   fwrite(&value,sizeof(value),1,fp);

   fclose(fp);

}

void food\_menu(){

   printf("\n\n Welcome to YUV Hotel \n Menu Card\n");

   printf("\n S.No.   Item Price(Rs.)");

   printf("\n  1. Tea              15");

   printf("\n  2. Coffee             10");

   printf("\n  3. Veg Breakfast 1    60");

   printf("\n  4. Veg Breakfast 2    80");

   printf("\n  5. Veg Lunch          50");

   printf("\n  6. NV Lunch           70");

   printf("\n  7. Veg Dinner         100");

   printf("\n  8. NV Dinner          120");

   printf("\n \n Enjoy your Meal!");

}

void room\_menu(){

   printf("\n Welcome to YUV Rooms \n Tariff Card\n");

   printf("\n S.No.   Room Type     Rate(Rs.)");

   printf("\n  1. AC Rooms                2000");

   printf("\n  2. Non-AC Rooms            1400");

}

int food\_cost(){

puts("\nEnter you choice: ");

int c;

scanf("%d",&c);

switch(c){

case 1:

return 15;

case 2:

return 10;

case 3:

return 60;

case 4:

return 80;

case 5:

return 50;

case 6:

return 70;

case 7:

return 100;

case 8:

return 120;

default:

printf("Invalid Choice!");

return 0;

break;

}

}

void billing(int room, int r){

struct guest p;

int flag=0;

while(flag==0){

if(room>6 && room<1){

puts("Room not found\t Enter a valid room number: ");

scanf(" %d",&room);

}

else

flag=1;

}

FILE \*f=fopen("register.bin","r+");

while(!feof(f)){

fread(&p,sizeof(p),1,f);

if(p.room\_no==room){

p.bill+=r;

fseek(f,-1\*sizeof(p),SEEK\_CUR);

fwrite(&p,sizeof(p),1,f);

break;

}

}

fclose(f);

}

void room\_dealloc(int room){

FILE \*f=fopen("room.bin","rb");

FILE \*ft=fopen("temp.bin","wb");

struct room delete;

while(!feof(f)){

fread(&delete,sizeof(delete),1,f);

if(delete.room\_no!=room){

fwrite(&delete,sizeof(delete),1,ft);

}

else{

printf("\nRoom no. %d available now!",room);

}

}

remove("room.bin");

rename("temp.bin","room.bin");

fclose(f);

fclose(ft);

}

int check\_out(int room){

FILE \*f=fopen("register.bin","rb");

FILE \*ft=fopen("temp.bin","wb");

struct guest going;

int total\_bill;

while(!feof(f)){

fread(&going,sizeof(going),1,f);

if(going.room\_no!=room){

fwrite(&going,sizeof(going),1,ft);

}

else{

total\_bill=going.bill;

}

}

remove("register.bin");

rename("temp.bin","register.bin");

fclose(f);

fclose(ft);

room\_dealloc(room);

return(total\_bill);

}

int main(){

room\_menu();

int choice;

do{

choice=0;

puts("\n1) Check In");

puts("\n2) Food menu");

puts("\n3) Check out");

puts("\n0) Exit");

puts("\n5) Diagnose software");

puts("\nEnter your choice:");

scanf(" %d",&choice);

switch(choice){

case 1:

customer\_registration();

break;

case 2:

food\_menu();

printf("\nEnter your room number: ");

int room;

scanf(" %d",&room);

int cost=food\_cost();

billing(room,cost);

break;

case 3:

puts("\nEnter your room number: ");

int room2;

scanf(" %d",&room2);

int disp=check\_out(room2);

printf("Have a nice day! Your total bill is %d",disp);

break;

case 5:

puts("\nEnter a to clear all customer");

puts("\nEnter b to clear all rooms");

char ch;

scanf(" %c",&ch);

puts("\nSURE? This can't be reverted! press 0 to cancel or 1 to continue\n");

int temp;

scanf(" %d",&temp);

if(temp==1){

if(ch=='a')

customer\_initialize();

else if(ch=='b')

room\_initialize();

else

puts("\nInvalid choice!");

}

else

printf("\nOperation cancelled! Reverting Back...");

break;

}

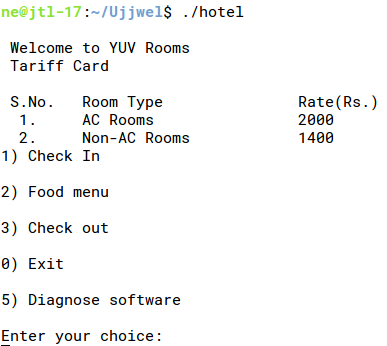
}while(choice);

}

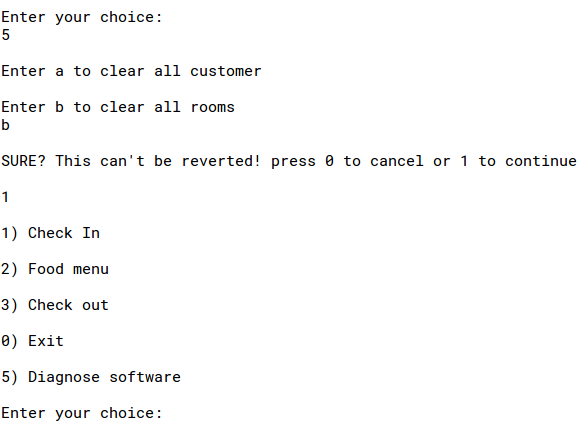
Compilation:

https://lh5.googleusercontent.com/rYYPQc94tMkSPM6kl467HTqw5kcMpduVV5J_9pR7q8AiNxrIH-PyGnK6bKajScYt3Rao61jbNZtlmyYvttcvYjjJeEcN5stbpTRW_HVY5-zBboRZgrJ2CQp4QupL9y6DtvYEtPMB

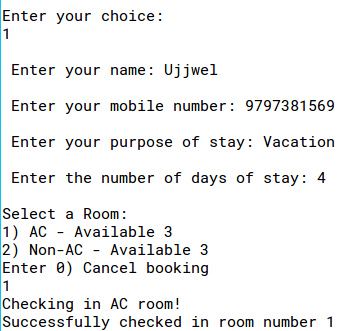
Welcome screen:

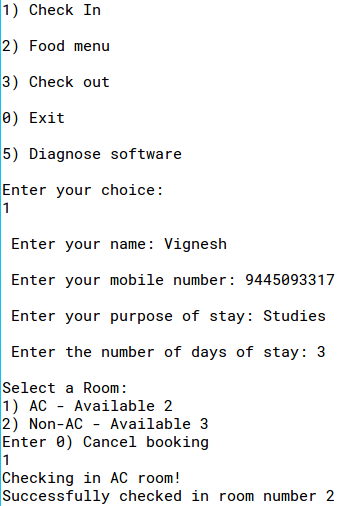


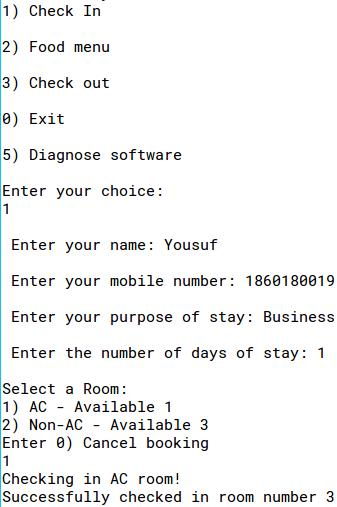
Initialization of the system:

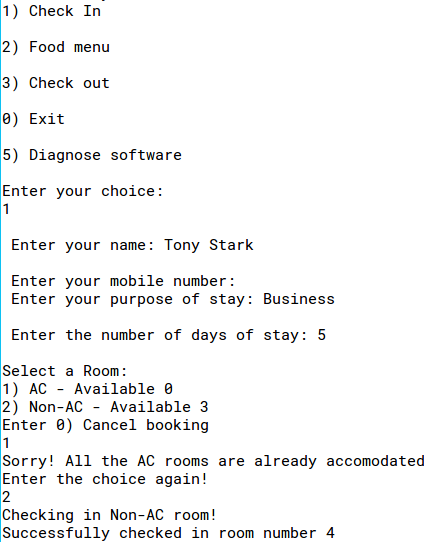


Check in:

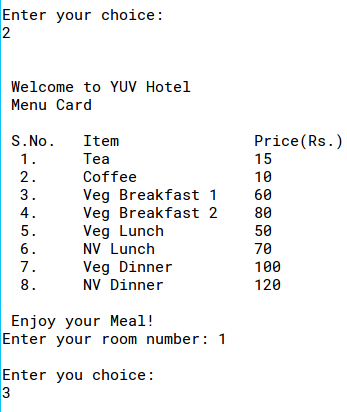


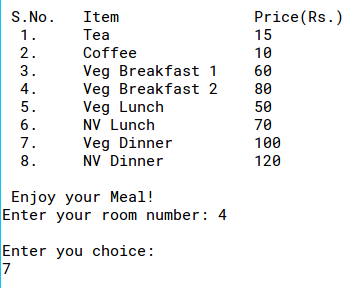




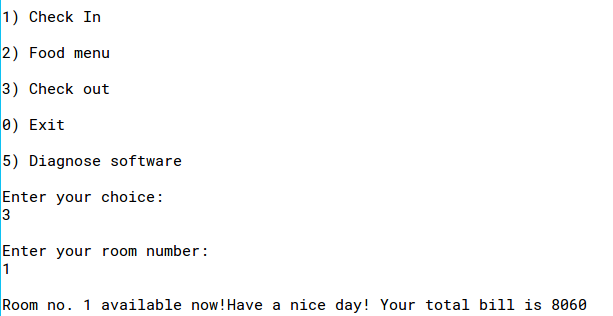


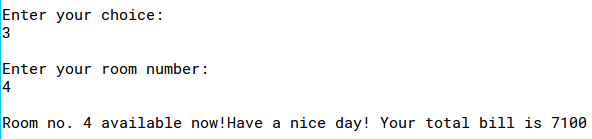
Ordering Food:





Check Out:





Terminating the program:

