C Programming

Project: Booking Hotel

29th November 2018

Name: Thirathapat Chaowai

ID: 61090037

13016235: C Programming

First Semester, 2018

Project Proposal

1. Project developer

Student ID	Name
61090037	Thirathapat Chaowai

2. Project title

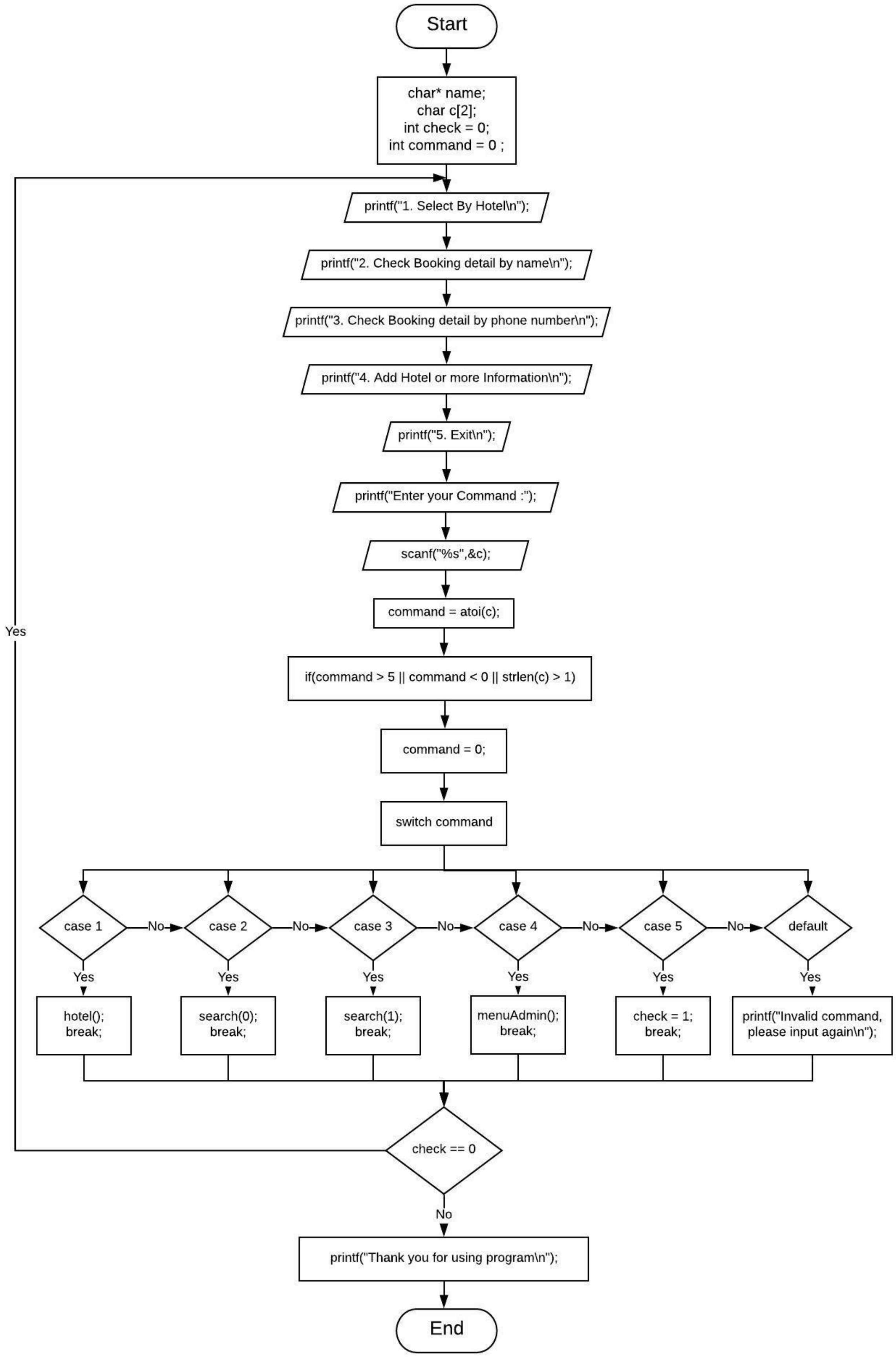
Booking hotel

3. Project description and requirements

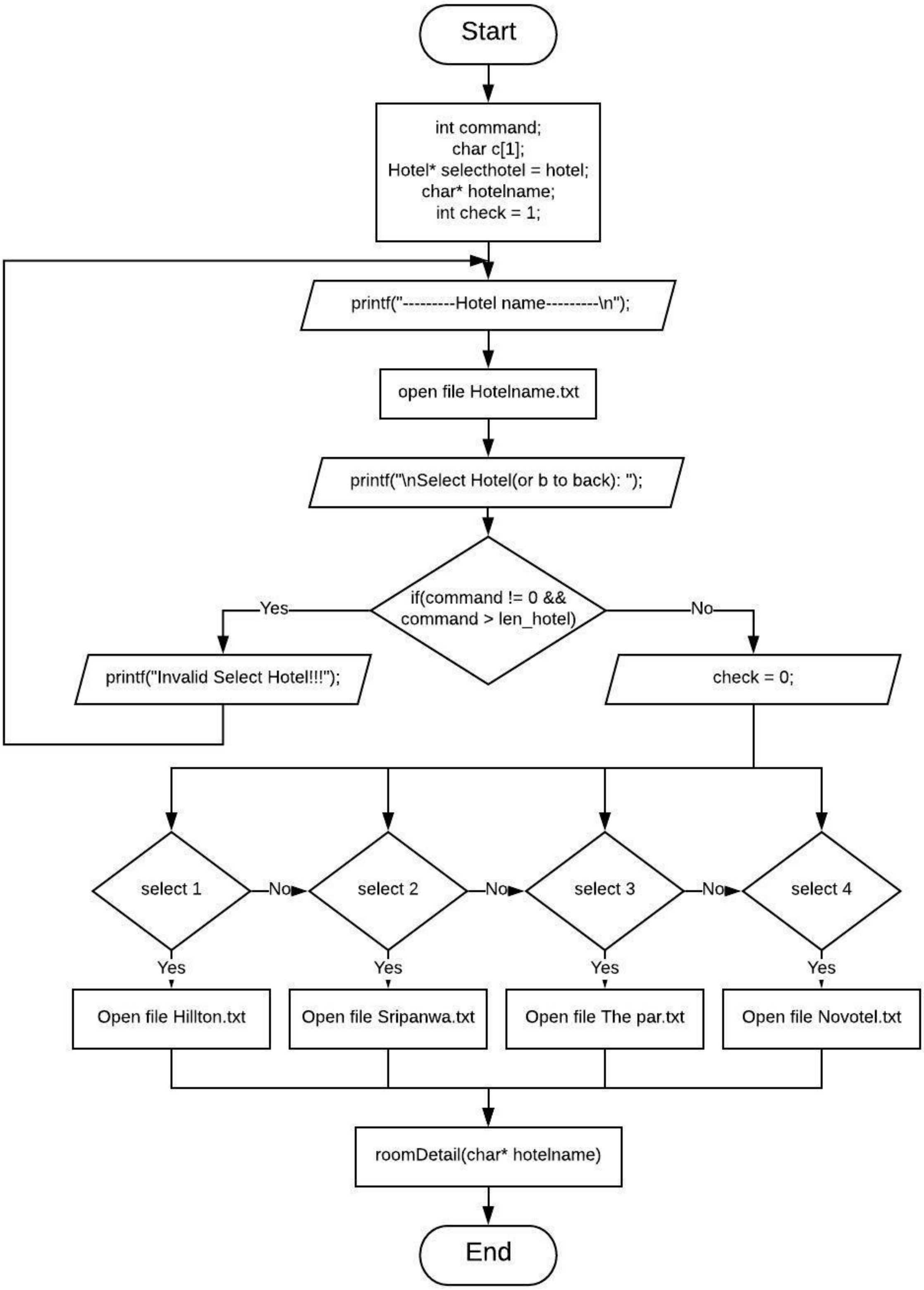
Project description:

Booking hotel is the program that suitable for people who want to stay comfortably. The program will allow user to prompt select hotel, check in and check out date and selects the type of room. The program also shows that number of available rooms. Other than, you can add your own hotel.

home()

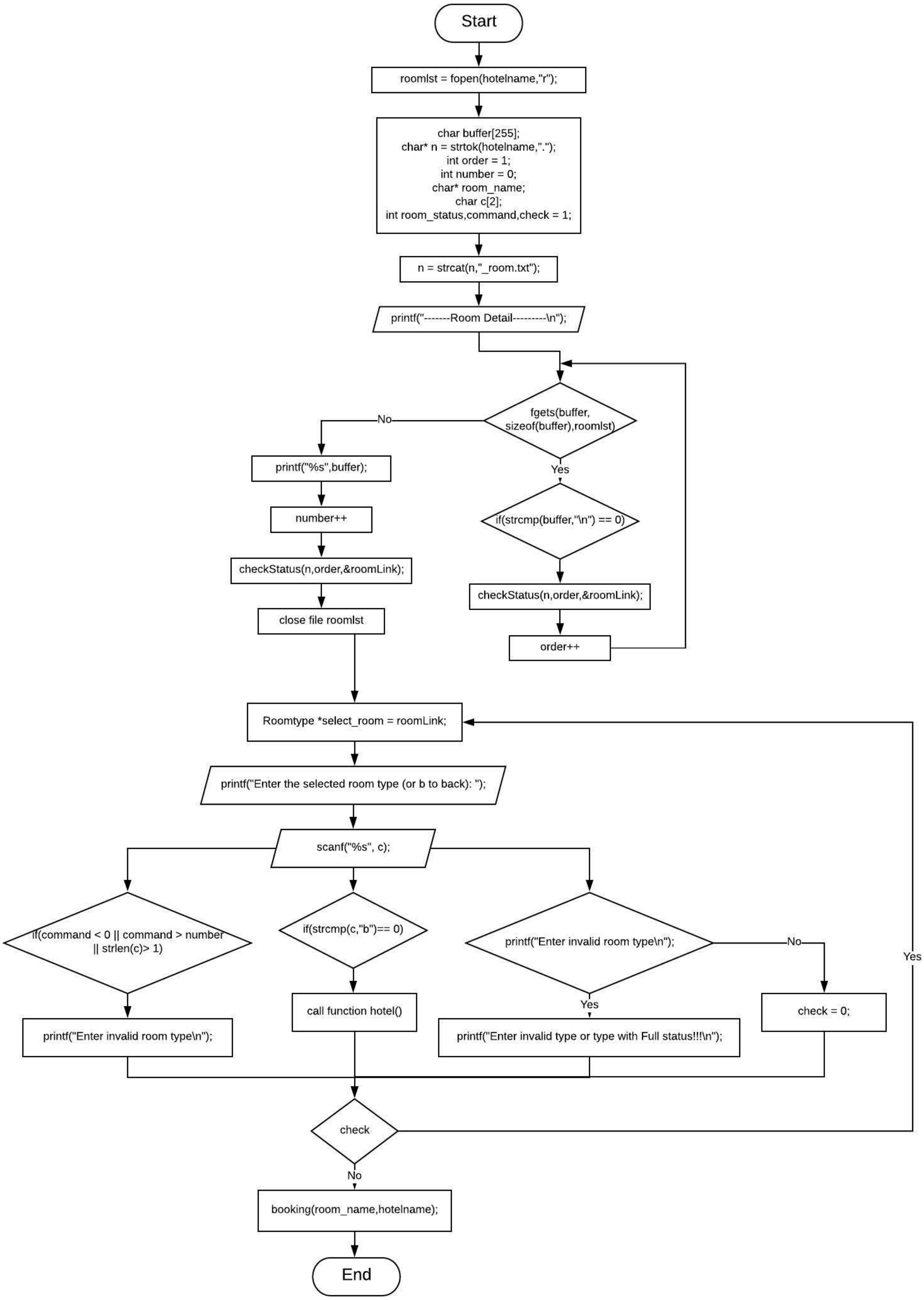

Start
char* name;
char c[2];
int check = 0;
int command = 0 ;
printf("1. Select By Hotel\n");
printf("2. Check Booking detail by name\n");
printf("3. Check Booking detail by phone number\n");
printf("4. Add Hotel or more Information\n");
printf("5. Exit\n");
printf("Enter your Command :");
scanf("%s",&c);
command = atoi(c);
if(command > 5 || command < 0 || strlen(c) > 1)
command = 0;
switch command
case 1
No
case 2
No
case 3
No
case 4
No
case 5
No
default
Yes
hotel();
break;
Yes
search(0);
break;
Yes
search(1);
break;
Yes
menuAdmin();
break;
Yes
check = 1;
break;
Yes
printf("Invalid command, please input again\n");
check == 0
Yes
No
printf("Thank you for using program\n");
End

hotel()

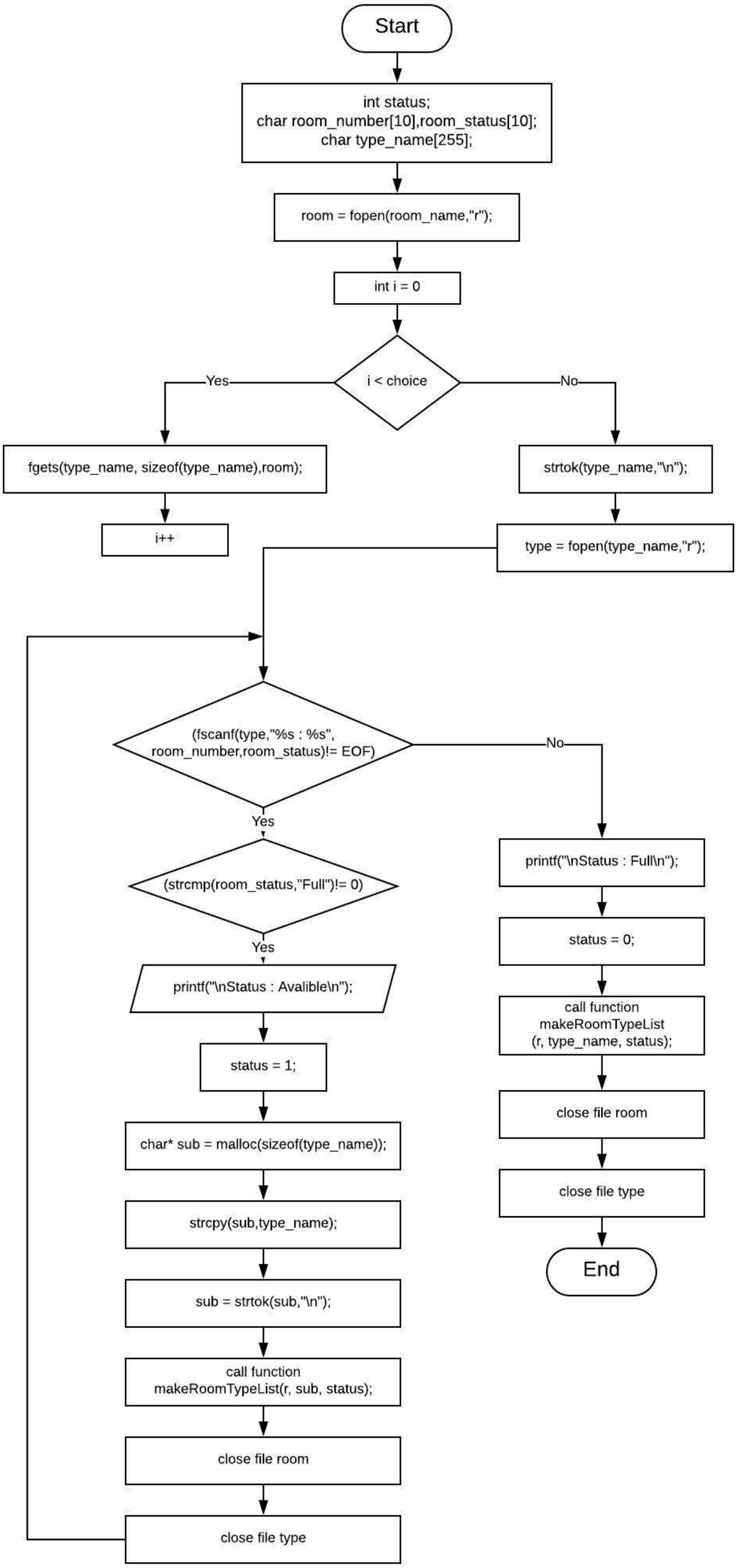


Start
int command;
char c[1];
Hotel* selecthotel = hotel;
char* hotelname;
int check = 1;
printf("---------Hotel name---------\n");
open file Hotelname.txt
printf("\nSelect Hotel(or b to back): ");
if(command != 0 &&
command > len_hotel)
Yes
No
printf("Invalid Select Hotel!!!");
check = 0;
select 1
select 2
select 3
select 4
No
No
No
Yes
Yes
Yes
Yes
Open file Hillton.txt
Open file Sripanwa.txt
Open file The par.txt
Open file Novotel.txt
roomDetail(char* hotelname)
End

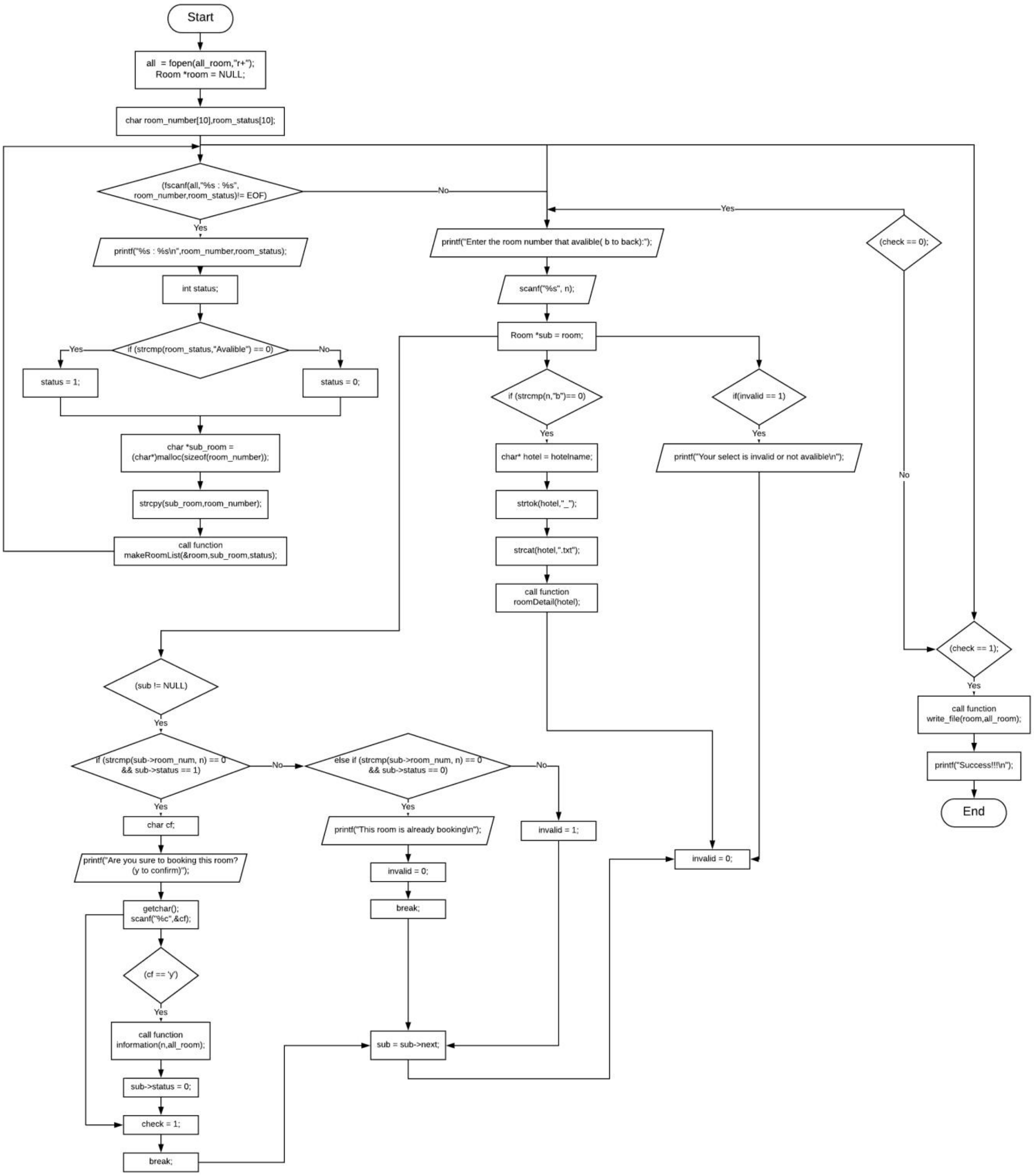
roomDetail()


Start
roomlst = fopen(hotelname,"r");
char buffer[255];
char* n = strtok(hotelname,".");
int order = 1;
int number = 0;
char* room_name;
char c[2];
int room_status,command,check = 1;
n = strcat(n,"_room.txt");
printf("-------Room Detail---------\n");
fgets(buffer, sizeof(buffer),roomlst)
No
Yes
printf("%s",buffer);
number++
checkStatus(n,order,&roomLink);
close file roomlst
if(strcmp(buffer,"\n") == 0)
checkStatus(n,order,&roomLink);
order++
Roomtype *select_room = roomLink;
printf("Enter the selected room type (or b to back): ");
scanf("%s", c);
if(command < 0 || command > number || strlen(c)> 1)
if(strcmp(c,"b")== 0)
printf("Enter invalid room type\n");
No
Yes
printf("Enter invalid room type\n");
call function hotel()
printf("Enter invalid type or type with Full status!!!\n");
check = 0;
Yes
check
No
booking(room_name,hotelname);
End

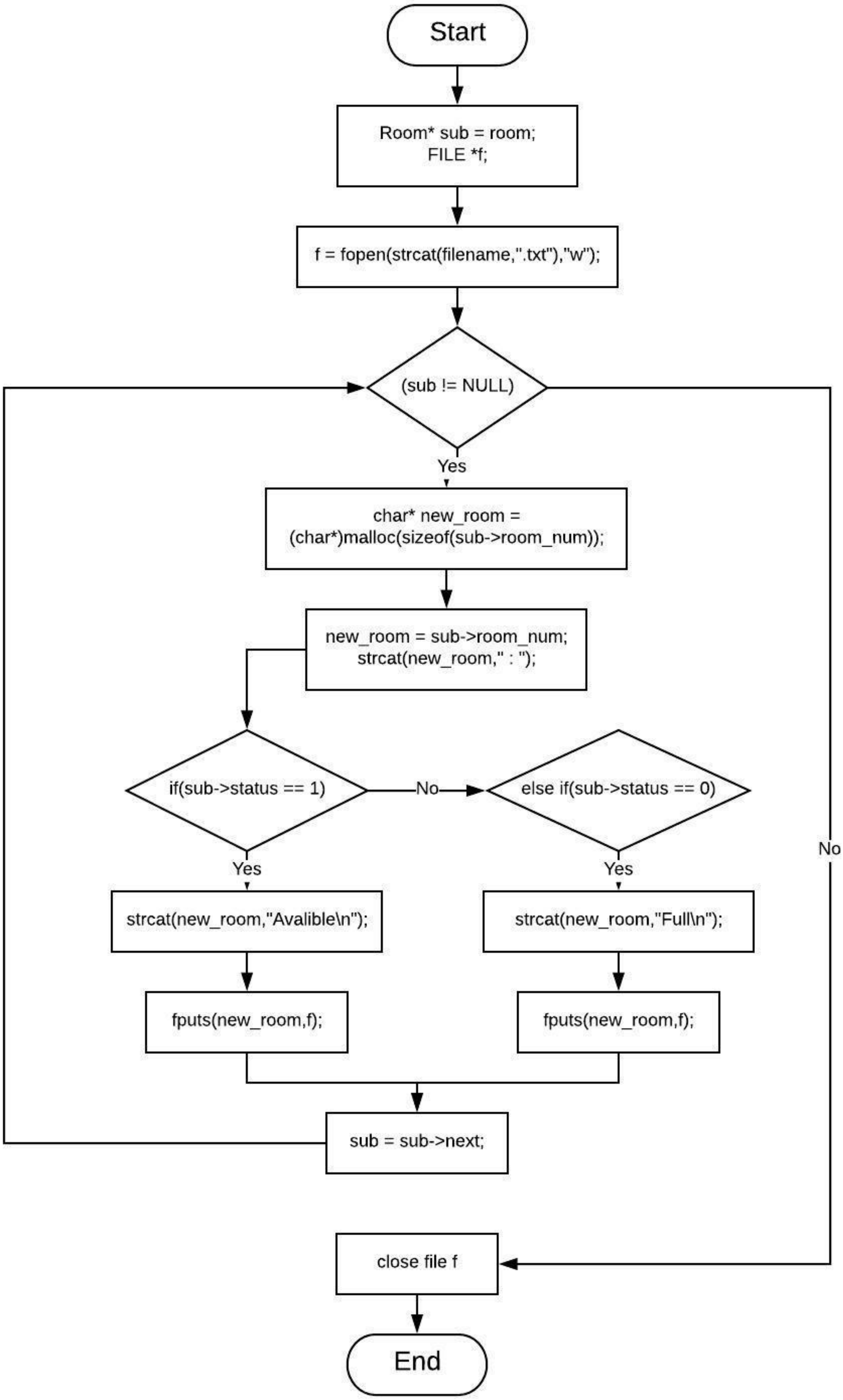
checkStatus()


Start
int status;
char room_number[10],room_status[10];
char type_name[255];
room = fopen(room_name,"r");
int i = 0
i < choice
Yes
No
fgets(type_name, sizeof(type_name),room);
i++
strtok(type_name,"\n");
type = fopen(type_name,"r");
(fscanf(type,"%s : %s",
room_number,room_status)!= EOF)
No
Yes
(strcmp(room_status,"Full")!= 0)
Yes
printf("\nStatus : Avalible\n");
status = 1;
char* sub = malloc(sizeof(type_name));
strcpy(sub,type_name);
sub = strtok(sub,"\n");
call function
makeRoomTypeList(r, sub, status);
close file room
close file type
printf("\nStatus : Full\n");
status = 0;
call function
makeRoomTypeList
(r, type_name, status);
close file room
close file type
End

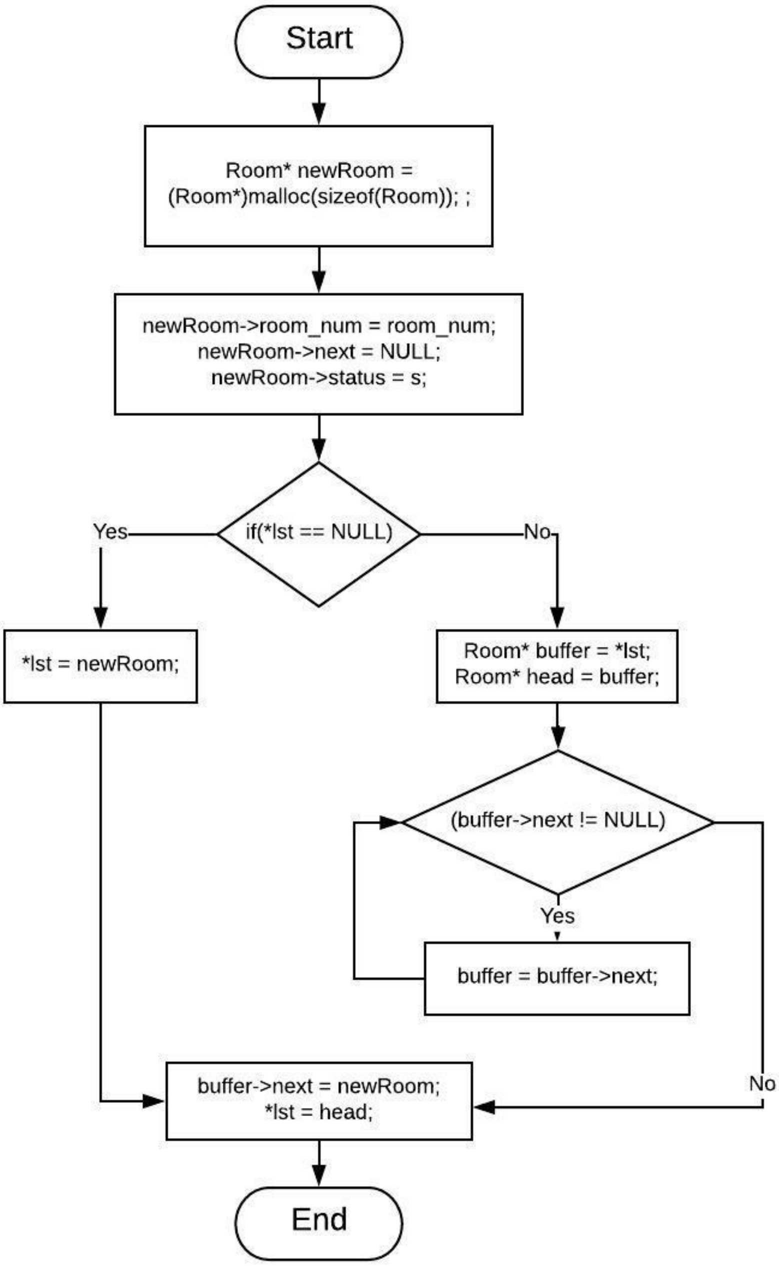
Booking()


Start
all = fopen(all_room,"r+");
Room *room = NULL;
char room_number[10],room_status[10];
(fscanf(all,"%s : %s",
room_number,room_status)!= EOF)
No
Yes
printf("%s : %s\n",room_number,room_status);
int status;
if (strcmp(room_status,"Avalible") == 0)
Yes
No
status = 1;
status = 0;
char *sub_room =
(char*)malloc(sizeof(room_number));
strcpy(sub_room,room_number);
call function
makeRoomList(&room,sub_room,status);
Yes
(check == 0);
No
printf("Enter the room number that avalible(b to back):");
scanf("%s", n);
Room *sub = room;
if (strcmp(n,"b")== 0)
Yes
char* hotel = hotelname;
strtok(hotel,"_");
strcat(hotel,".txt");
call function
roomDetail(hotel);
if(invalid == 1)
Yes
printf("Your select is invalid or not avalible\n");
(check == 1);
Yes
call function
write_file(room,all_room);
printf("Success!!!\n");
End
(sub != NULL)
Yes
if (strcmp(sub->room_num, n) == 0
&& sub->status == 1)
No
else if (strcmp(sub->room_num, n) == 0
&& sub->status == 0)
No
Yes
char cf;
printf("Are you sure to booking this room?
(y to confirm)");
getchar();
scanf("%c",&cf);
(cf == 'y')
Yes
call function
information(n,all_room);
sub->status = 0;
check = 1;
break;
Yes
printf("This room is already booking\n");
invalid = 0;
break;
invalid = 1;
sub = sub->next;
invalid = 0;

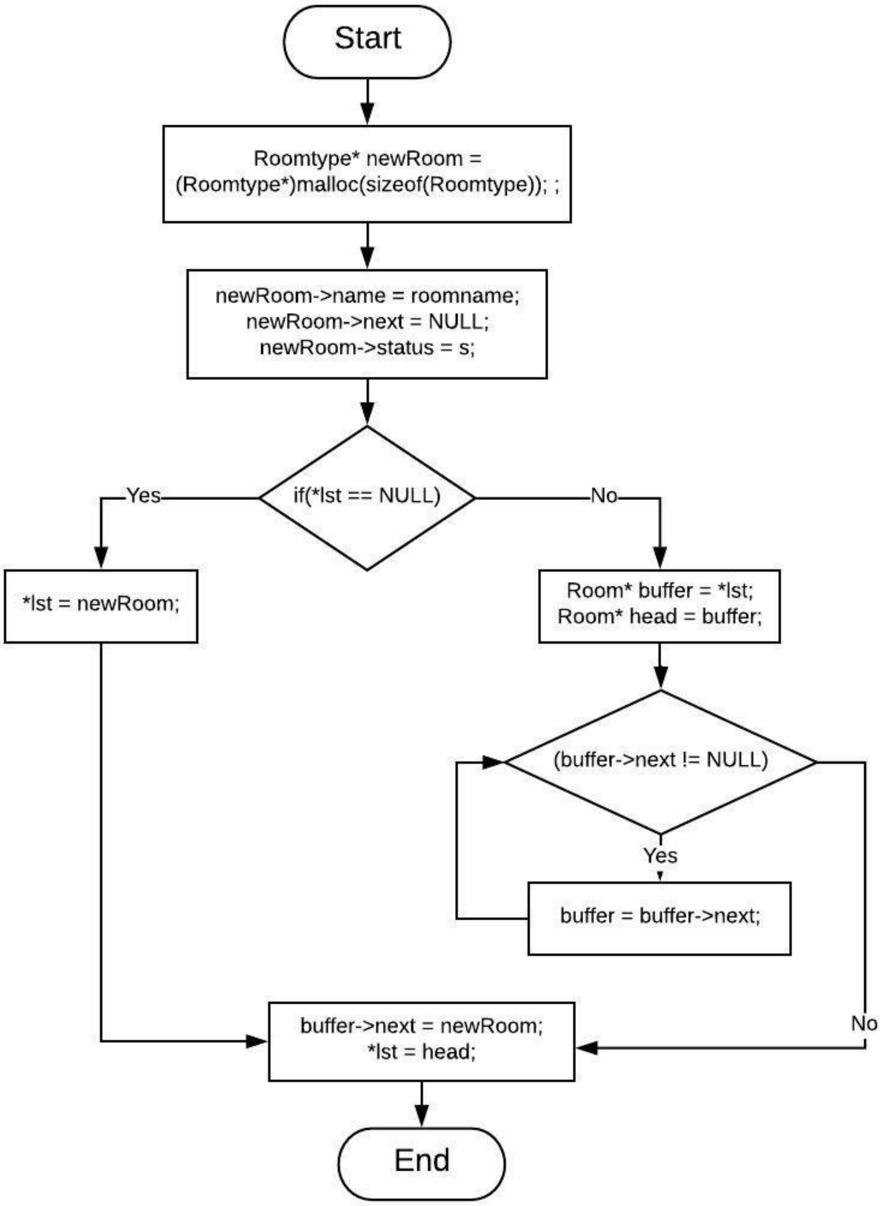
write_file()


Start
Room* sub = room;
FILE *f;
f = fopen(strcat(filename,".txt"),"w");
(sub != NULL)
Yes
char* new_room =
(char*)malloc(sizeof(sub->room_num));
new_room = sub->room_num;
strcat(new_room," : ");
if(sub->status == 1)
No
else if(sub->status == 0)
Yes
Yes
strcat(new_room,"Avalible\n");
strcat(new_room,"Full\n");
fputs(new_room,f);
fputs(new_room,f);
sub = sub->next;
No
close file f
End

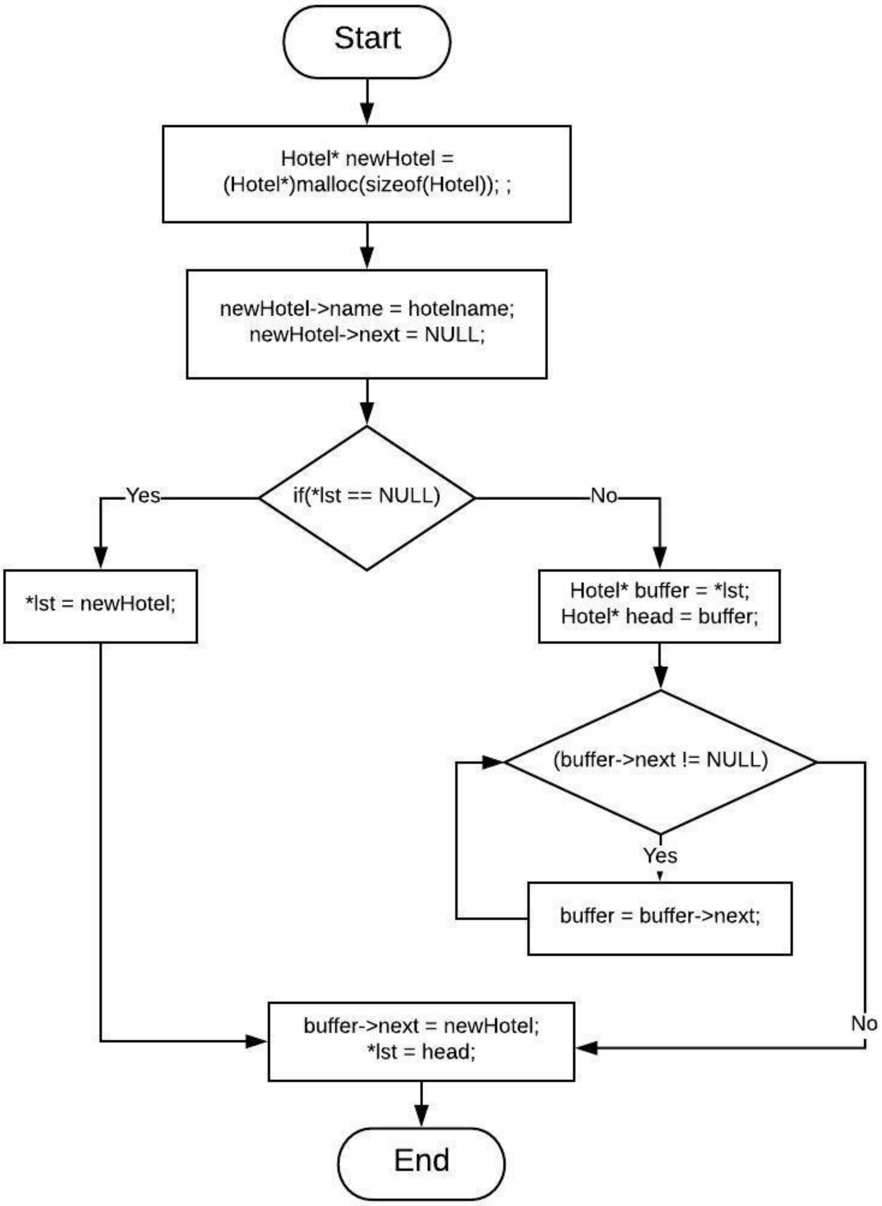
makeRoomList()



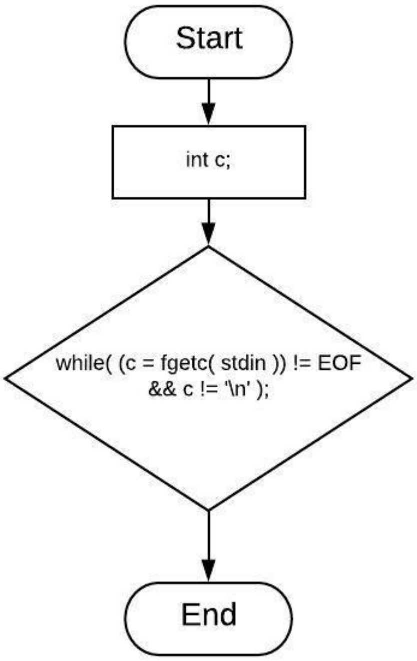
makeRoomTypeList()



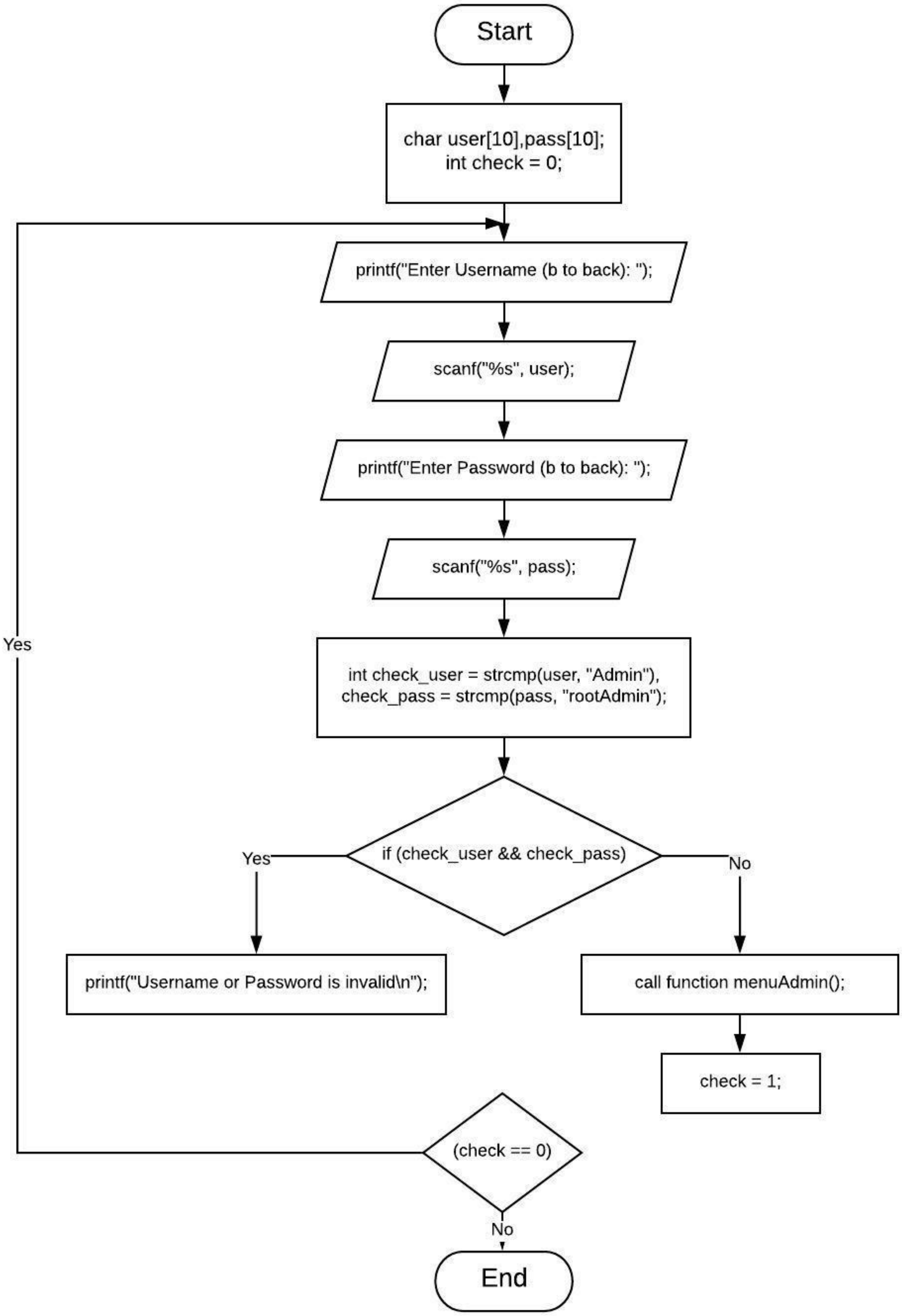
makeHotelList()



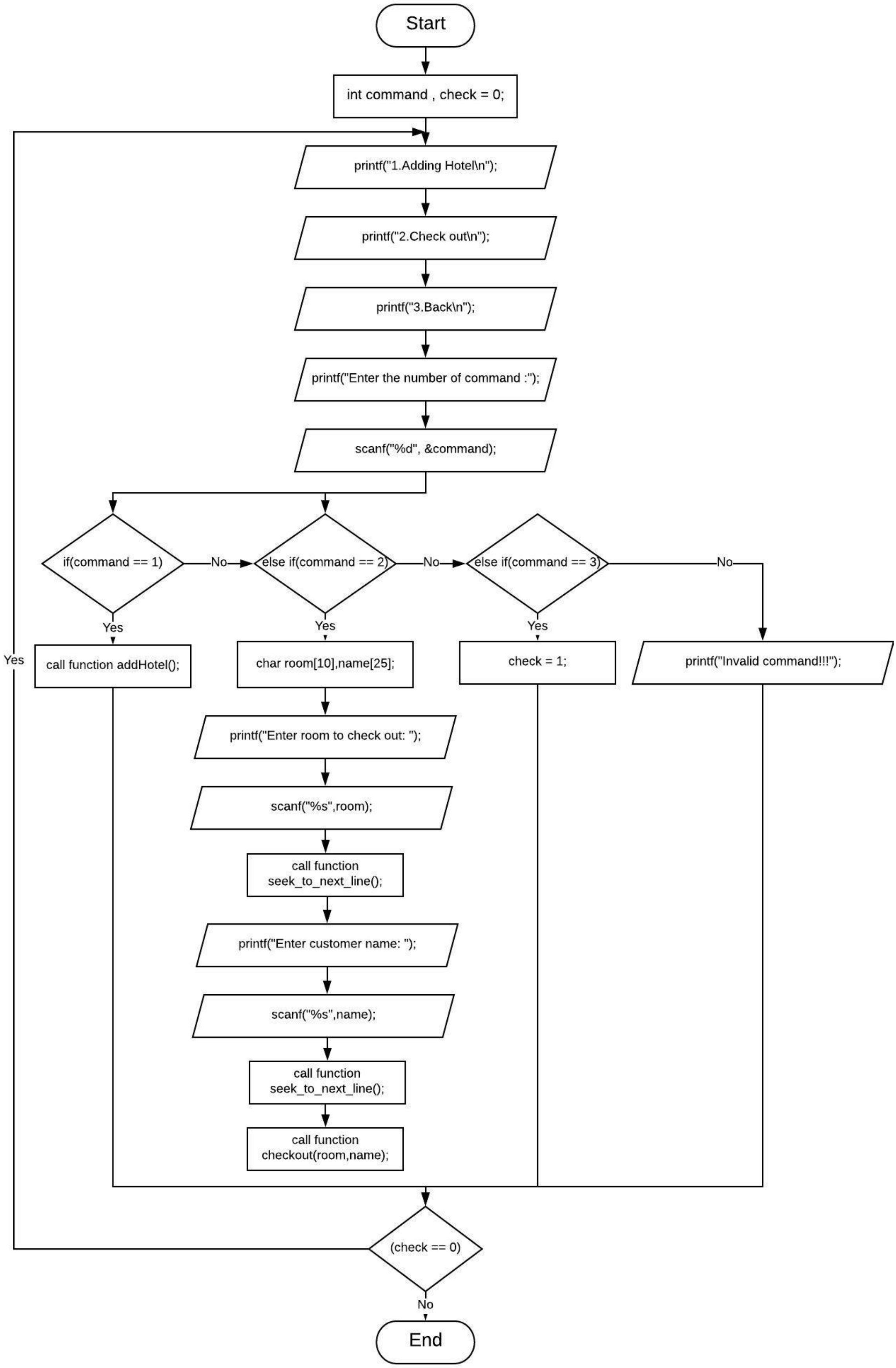
Seek_to_next_line()



login()

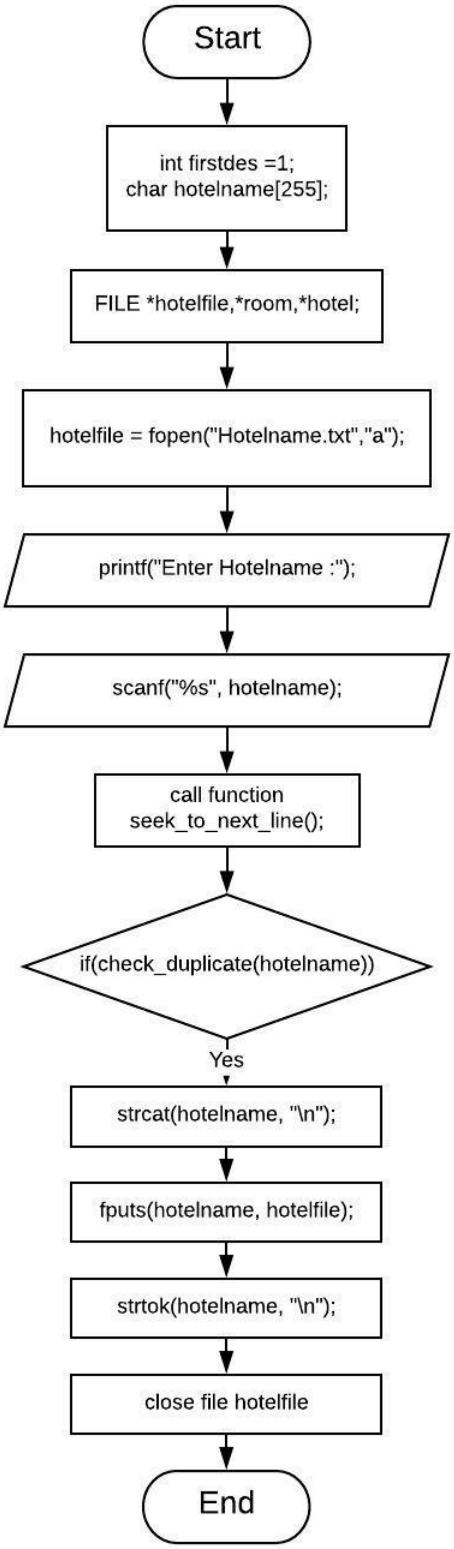

Start
char user[10],pass[10];
int check = 0;
printf("Enter Username (b to back): ");
scanf("%s", user);
printf("Enter Password (b to back): ");
scanf("%s", pass);
int check_user = strcmp(user, "Admin"),
check_pass = strcmp(pass, "rootAdmin");
if (check_user && check_pass)
Yes
No
printf("Username or Password is invalid\n");
call function menuAdmin();
check = 1;
(check == 0)
Yes
No
End

manuAdmin()

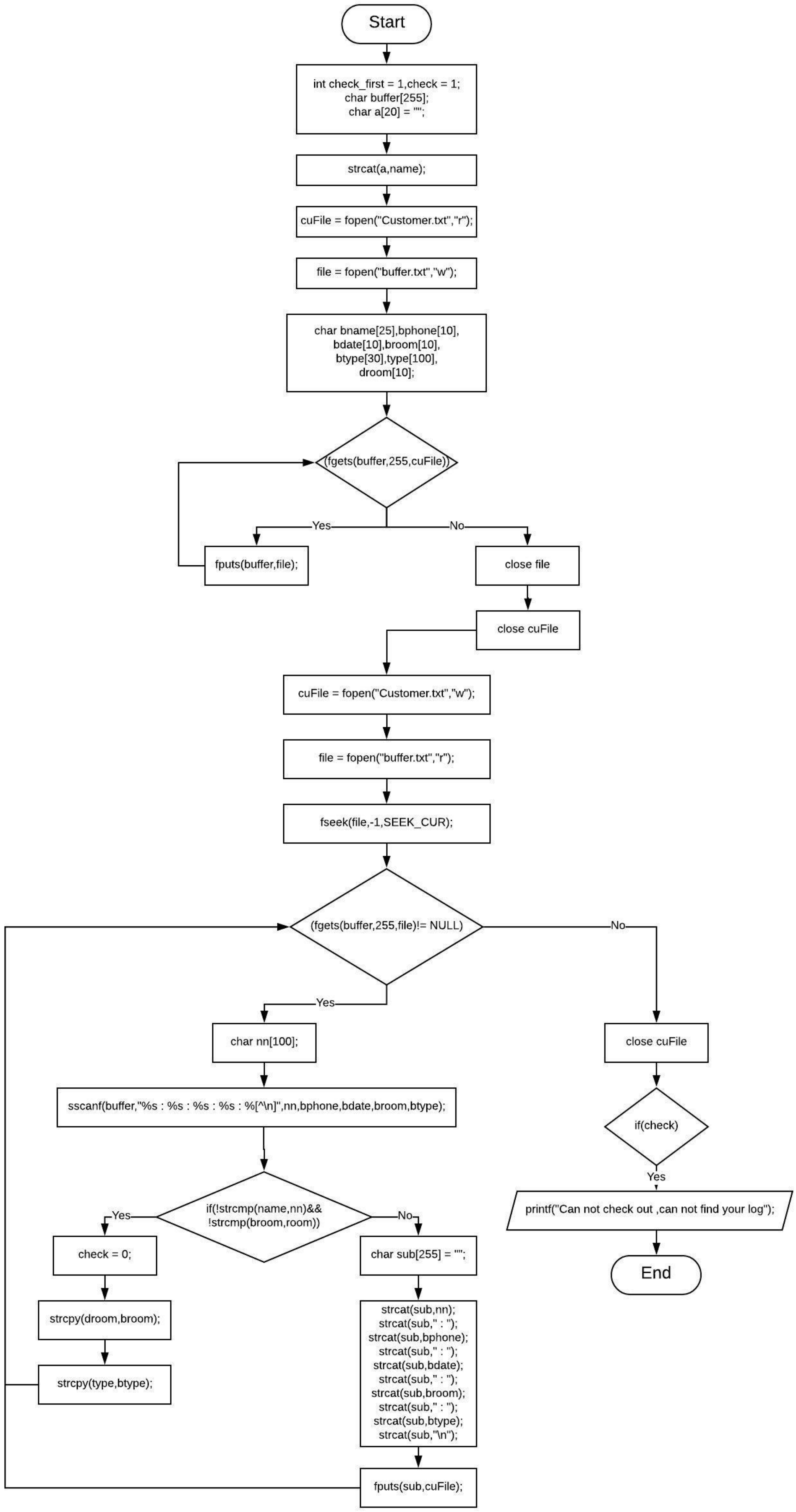


Start
int command , check = 0;
printf("1.Adding Hotel\n");
printf("2.Check out\n");
printf("3.Back\n");
printf("Enter the number of command :");
scanf("%d", &command);
if(command == 1)
No
else if(command == 2)
No
else if(command == 3)
No
Yes
call function addHotel();
Yes
char room[10],name[25];
printf("Enter room to check out: ");
scanf("%s",room);
call function seek_to_next_line();
printf("Enter customer name: ");
scanf("%s",name);
call function seek_to_next_line();
call function checkout(room,name);
Yes
check = 1;
printf("Invalid command!!!");
Yes
(check == 0)
No
End

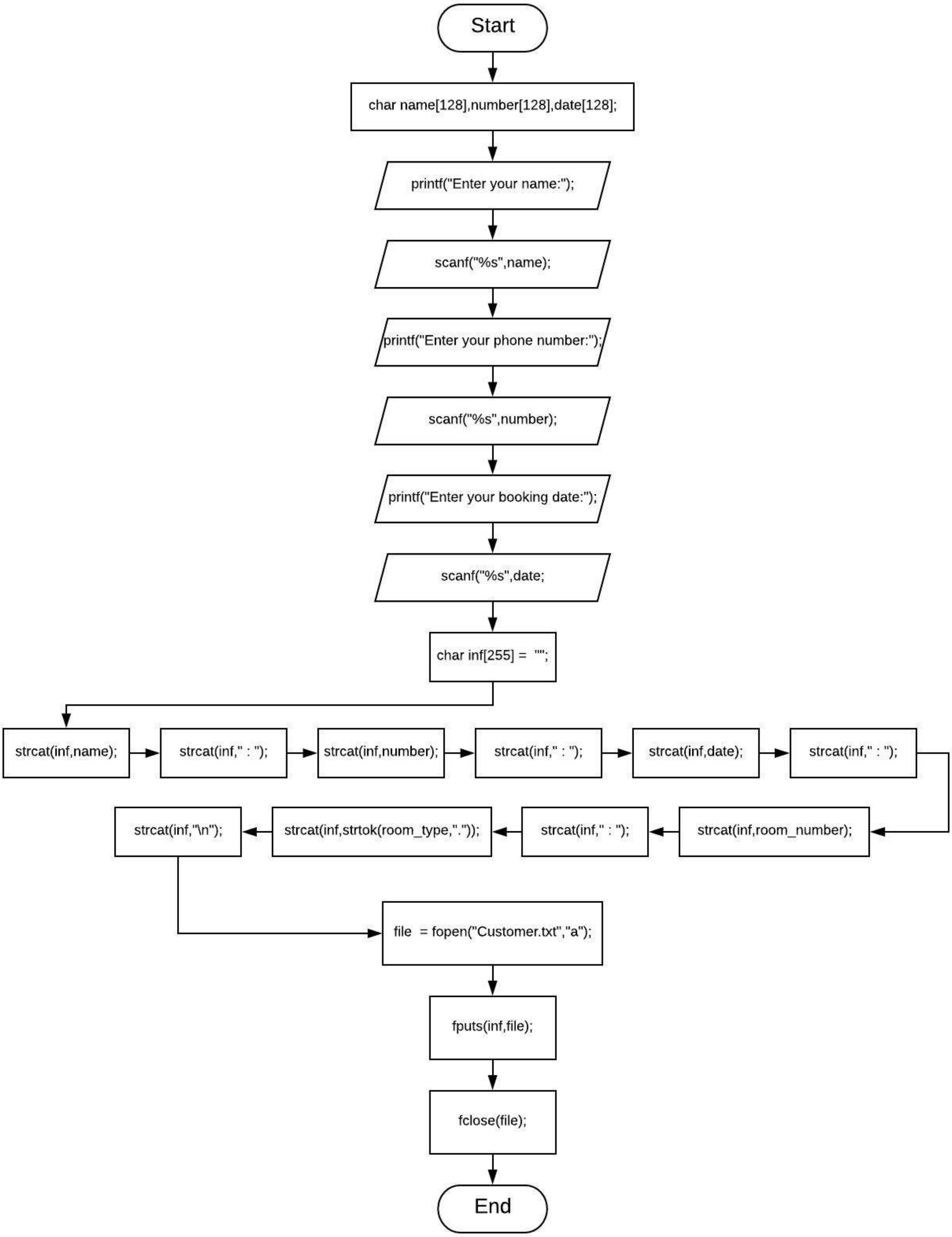
addHotel()


Start
int firstdes =1;
char hotelname[255];
FILE *hotelfile,*room,*hotel;
hotelfile = fopen("Hotelname.txt","a");
printf("Enter Hotelname :");
scanf("%s", hotelname);
call function
seek_to_next_line();
if(check_duplicate(hotelname))
Yes
strcat(hotelname, "\n");
fputs(hotelname, hotelfile);
strtok(hotelname, "\n");
close file hotelfile
End

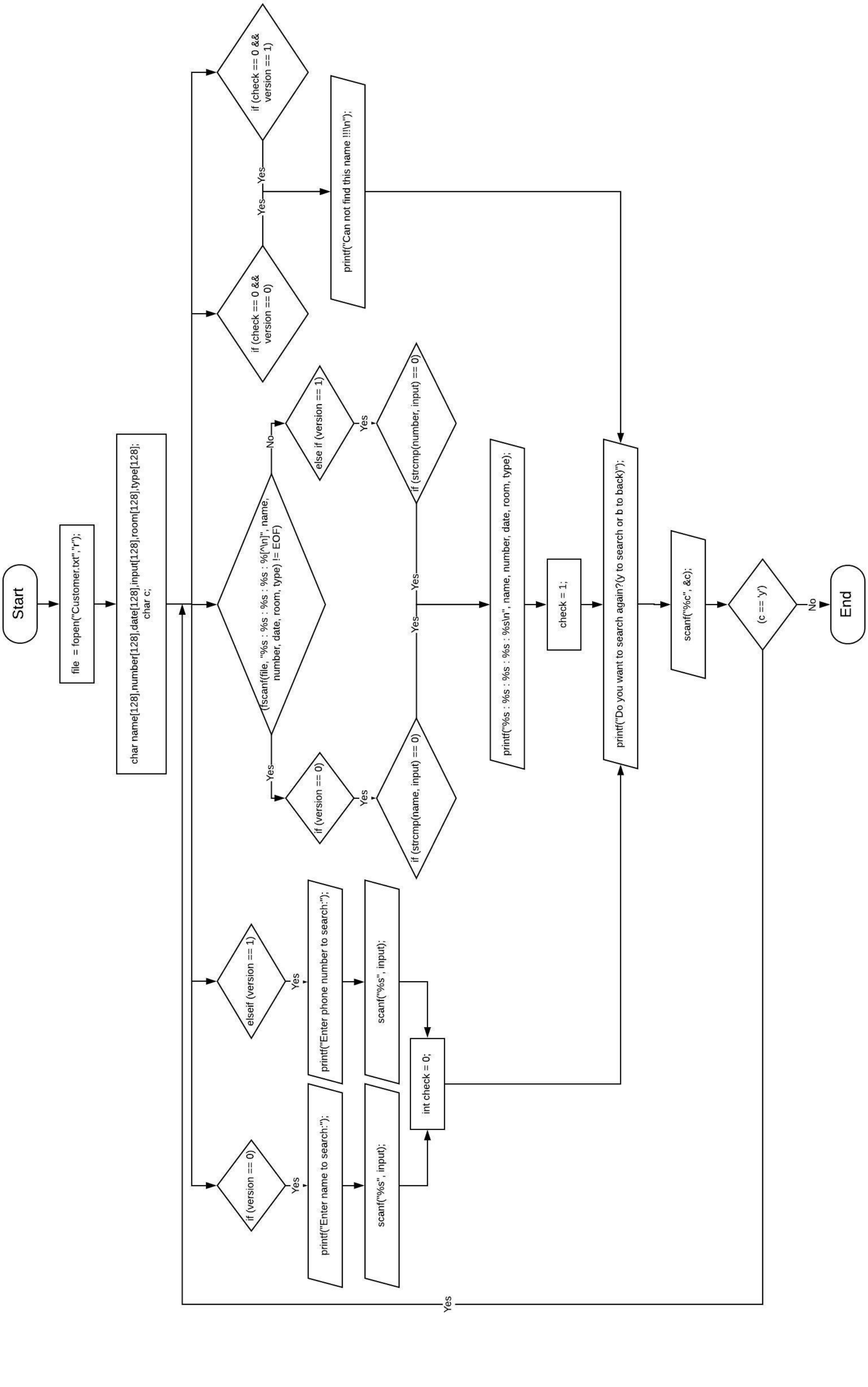
checkout()


Start
int check_first = 1,check = 1;
char buffer[255];
char a[20] = "";
strcat(a,name);
cuFile = fopen("Customer.txt","r");
file = fopen("buffer.txt","w");
char bname[25],bphone[10],
bdate[10],broom[10],
btype[30],type[100],
droom[10];
(fgets(buffer,255,cuFile))
Yes
No
fputs(buffer,file);
close file
close cuFile
cuFile = fopen("Customer.txt","w");
file = fopen("buffer.txt","r");
fseek(file,-1,SEEK_CUR);
(fgets(buffer,255,file)!= NULL)
No
Yes
char nn[100];
sscanf(buffer,"%s : %s : %s : %s : %[^\n]",nn,bphone,bdate,broom,btype);
if(!strcmp(name,nn)&&
!strcmp(broom,room))
Yes
No
check = 0;
strcpy(droom,broom);
strcpy(type,btype);
char sub[255] = "";
strcat(sub,nn);
strcat(sub," : ");
strcat(sub,bphone);
strcat(sub," : ");
strcat(sub,bdate);
strcat(sub," : ");
strcat(sub,broom);
strcat(sub," : ");
strcat(sub,btype);
strcat(sub,"\n");
fputs(sub,cuFile);
close cuFile
if(check)
Yes
printf("Can not check out ,can not find your log");
End

information()



search()


Start
file = fopen("Customer.txt","r");
char name[128],number[128],date[128],input[128],room[128],type[128];
char c;
if (version == 0)
Yes
printf("Enter name to search:");
scanf("%s", input);
elseif (version == 1)
Yes
printf("Enter phone number to search:");
scanf("%s", input);
int check = 0;
(fscanf(file, "%s : %s : %s : %[^\n]", name, number, date, room, type) != EOF)
Yes
if (version == 0)
Yes
if (strcmp(name, input) == 0)
Yes
No
else if (version == 1)
Yes
if (strcmp(number, input) == 0)
Yes
printf("%s : %s : %s : %s\n", name, number, date, room, type);
check = 1;
if (check == 0 && version == 0)
Yes
if (check == 0 && version == 1)
Yes
printf("Can not find this name !!!\n");
printf("Do you want to search again?(y to search or b to back)");
scanf("%c", &c);
(c == 'y')
No
End
Yes

Main.c

```
#include <stdio.h>

#include <string.h>

#include "Hotel.h"

#include "Admin.h"

#include "User.h"

void home(){

    char* name;

    char c[2];

    int check = 0;

    printf("|---------- Welcome to booking hotel ----------|\n\n");

    do{

        int command = 0 ;

        printf("1. Select By Hotel\n");

        printf("2. Check Booking detail by name\n");

        printf("3. Check Booking detail by phone number\n");

        printf("4. Admin menu\n");

        printf("5. Exit\n");

        printf("Enter your Command :");

        scanf("%s",&c);

        seek_to_next_line();

        command = atoi(c);

        if(command > 5 || command < 0 || strlen(c) > 1){

            command = 0 ;

        }

        switch (command){

            case 1:

                printf("\n");hotel();
                break;
            case 2:
                printf("\n");search(0);
                break;
            case 3:
                printf("\n");search(1);
                break;
            case 4:
                printf("\n");login();  //Username  = Admin , Pass = rootAdmin
                break;
            case 5:
                printf("\n");check = 1;
                break;
            default:
                printf("Invalid command, please input again\n");
        }
    }while(check == 0 );
    printf("Thank you for using program\n");
}

int main(int argc, char* argv[]) {
    home();}
```

Hotel.h

```
#ifndef BOOKINGHOTEL_HOTEL_H
#define BOOKINGHOTEL_HOTEL_H

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "User.h"

void roomDetail(char* hotelname);
void hotel();

typedef struct hotel {
    char* name;
    struct hotel * next;
}Hotel;

typedef  struct roomtype{
    char *name;
    int status;
    struct roomtype *next;
}Roomtype;

typedef  struct room{
    char *room_num;
    int status;
    struct room *next;
}Room;
```

```
void makeHotelList(Hotel **lst, char *hotelname){
    Hotel* newHotel = (Hotel*)malloc(sizeof(Hotel)); ;
    newHotel->name = hotelname;
    newHotel->next = NULL;
    if(*lst == NULL) {
        *lst = newHotel;
    }
    else{
        Hotel* buffer = *lst;
        Hotel* head = buffer;
        while(buffer->next != NULL) {
            buffer = buffer->next;
        }
        buffer->next = newHotel;
        *lst = head;
    }
}

void makeRoomTypeList(Roomtype **lst, char *roomname, int s){
    Roomtype* newRoom = (Roomtype*)malloc(sizeof(Roomtype)); ;
    newRoom->name = roomname;
    newRoom->next = NULL;
    newRoom->status = s;

    if(*lst == NULL) {
        *lst = newRoom;
    }
    else{
        Roomtype* buffer = *lst;
        Roomtype* head = buffer;
        while(buffer->next != NULL) {
            buffer = buffer->next;
        }
        buffer->next = newRoom;
        *lst = head;
    }
}

void makeRoomList(Room **lst, char *room_num, int s){
    Room* newRoom = (Room*)malloc(sizeof(Room)); ;
    newRoom->room_num = room_num;
    newRoom->next = NULL;
    newRoom->status = s;

    if(*lst == NULL) {
        *lst = newRoom;
    }
    else{
        Room* buffer = *lst;
        Room* head = buffer;
        while(buffer->next != NULL) {
            buffer = buffer->next;
        }
        buffer->next = newRoom;
        *lst = head;
    }
}

void write_file(Room* room,char* filename){
    Room* sub = room;
    FILE *f;
    f = fopen(strcat(filename,".txt"),"w");

    while (sub != NULL) {
        char* new_room = (char*)malloc(sizeof(sub->room_num));
        new_room = sub->room_num;
        strcat(new_room," : ");
        if(sub->status == 1){
            strcat(new_room,"Avalible\n");
            fputs(new_room,f);
        }
        else if(sub->status == 0){
            strcat(new_room,"Full\n");
            fputs(new_room,f);
        }
        sub = sub->next;
    }
    fclose(f);
}

void booking(char* all_room, char* hotelname){
    FILE * all;
```

```
all  = fopen(all_room,"r+");

Room *room = NULL;

char room_number[10],room_status[10];


while (fscanf(all,"%s : %s",room_number,room_status)!= EOF){

    printf("%s : %s\n",room_number,room_status);

    int status;

    if(strcmp(room_status,"Avalible") == 0)

        status = 1;

    else

        status = 0;


    char *sub_room = (char*)malloc(sizeof(room_number));

    strcpy(sub_room,room_number);

    makeRoomList(&room,sub_room,status);

}


char n[10];

int check = 0, invalid = 0;




do {

    printf("Enter the room number that avalible( b to back):");

    scanf("%s", n);

    Room *sub = room;

    if(strcmp(n,"b")== 0){

        char* hotel = hotelname;

        strtok(hotel,"_");
        strcat(hotel,".txt");
        roomDetail(hotel);
        return;
    }
    while (sub != NULL) {
        if (strcmp(sub->room_num, n) == 0 && sub->status == 1) {
            char cf;
            printf("Are you sure to booking this room?(y to confirm)");
            getchar();
            scanf("%c",&cf);
            if(cf == 'y'){
                information(n,all_room);
                sub->status = 0;
            }
            check = 1;
            break;
        }
        else if (strcmp(sub->room_num, n) == 0 && sub->status == 0){
            printf("This room is already booking\n");
            invalid = 0;
            break;
        }
        else
            invalid = 1;

        sub = sub->next;
    }
    if(invalid == 1)
```

```
            printf("Your select is invalid or not avalible\n");

        invalid = 0;

    }while (check == 0);

    if(check == 1){
        write_file(room,all_room);
        printf("Success!!!\n");
    }
}

void checkStatus(char* room_name,int choice,Roomtype **r){
    FILE *room,*type;
    int status;
    char room_number[10],room_status[10];

    room = fopen(room_name,"r");
    char type_name[255];
    for(int i =0;i< choice;i++){
        fgets(type_name, sizeof(type_name),room);
    }
    strtok(type_name,"\n");

    type = fopen(type_name,"r");
    while (fscanf(type,"%s : %s",room_number,room_status)!= EOF){
        if(strcmp(room_status,"Full")!= 0){
            printf("\nStatus : Avalible\n");
            status = 1;

            char* sub = malloc(sizeof(type_name));

            strcpy(sub,type_name);

            sub = strtok(sub,"\n");

            makeRoomTypeList(r, sub, status);

            fclose(room);

            fclose(type);

            return;

        }

    }

    printf("\nStatus : Full\n");

    status = 0;

    makeRoomTypeList(r, type_name, status);

    fclose(room);

    fclose(type);

}



void roomDetail(char* hotelname){

    FILE *roomlst;

    roomlst = fopen(hotelname,"r");

    if(roomlst == 0){

        return;

    }

    Roomtype *roomLink = NULL;


    char buffer[255];

    char* n = strtok(hotelname,".");
```

```
n = strcat(n,"_room.txt");


int order = 1;

int number = 0;


printf("-------Room Detail---------\n");

while (fgets(buffer, sizeof(buffer),roomlst)){

    if(strcmp(buffer,"\n") == 0){

        checkStatus(n,order,&roomLink);

        order++;

    }

    printf("%s",buffer);

    number++;

}

checkStatus(n,order,&roomLink);


fclose(roomlst);


char* room_name;

char c[2];

int room_status,command,check = 1;


do {

    Roomtype *select_room = roomLink;

    printf("\nEnter the selected room type (or b to back): ");

    scanf("%s", c);

    if(strcmp(c,"b")== 0){

        hotel();
```

```
            return;
        }
        command = atoi(c);
        if(command < 0 || command > number || strlen(c)> 1){
            printf("Enter invalid room type\n");
            continue;
        }

        for (int i = 0; i < command; i++) {
            room_name = select_room->name;
            room_status = select_room->status;
            select_room = select_room->next;
        }
        if (!room_status)
            printf("Enter invalid type or type with Full status!!!\n");
        else
            check = 0;

    }while(check);

    booking(room_name,hotelname);

}

void hotel() {
    FILE *hotelst;
    char buffer[255];
    Hotel *hotel = NULL;
```

```
int len_hotel = 0;

printf("\n---------Hotel name---------\n");
hotelst = fopen("Hotelname.txt","r");
while (fgets(buffer, sizeof(buffer),hotelst)){
  printf("%d.%s",len_hotel+1,buffer);
  char* sub = malloc(sizeof(buffer));
  strcpy(sub,buffer);
  sub = strtok(sub,"\n");
  makeHotelList(&hotel, sub);
  len_hotel++;
}

fclose(hotelst);

int command;
char c[1];
Hotel* selecthotel = hotel;
char* hotelname;
int check = 1;

do{
printf("\nSelect Hotel(or b to back): ");
scanf("%s",c);
if(strcmp(c,"b")== 0) {
  return;
}
command = atoi(c);
```

```
    if(command != 0 && command > len_hotel) printf("Invalid Select Hotel!!!");
    else check = 0;
    }while (check);


    for(int i = 0;i<command ;i++){
        hotelname = selecthotel->name;
        selecthotel = selecthotel->next;
    }


    hotelname = strcat(hotelname,".txt");
    roomDetail(hotelname);


}
#endif //BOOKINGHOTEL_HOTEL_H
```

Admin.h

```
#ifndef BOOKINGHOTEL_ADMIN_H

#define BOOKINGHOTEL_ADMIN_H




#include<stdio.h>

#include<string.h>

#include <stdlib.h>

#include <ctype.h>

#include "Hotel.h"




void seek_to_next_line( void )

{

   int c;

   while( (c = fgetc( stdin )) != EOF && c != '\n' );

}




void checkout(char room[10],char name[10]){

   FILE *cuFile,*file;

   int check_first = 1,check = 1;

   char buffer[255];

   char a[20] = "";

   strcat(a,name);

   cuFile = fopen("Customer.txt","r");

   file = fopen("buffer.txt","w");

   char bname[25],bphone[10],bdate[10],broom[10],btype[30],type[100],droom[10];
```

```
while(fgets(buffer,255,cuFile)){
    fputs(buffer,file);
}
fclose(file);
fclose(cuFile);
cuFile = fopen("Customer.txt","w");
file = fopen("buffer.txt","r");
fseek(file,-1,SEEK_CUR);
while(fgets(buffer,255,file)!= NULL){
    char nn[100];
    sscanf(buffer,"%s : %s : %s : %s : %[^\n]",nn,bphone,bdate,broom,btype);
    if(!strcmp(name,nn)&& !strcmp(broom,room)){
        check = 0;
        strcpy(droom,broom);
        strcpy(type,btype);
        continue;
    }
    else{
        char sub[255] = "";
        strcat(sub,nn);
        strcat(sub," : ");
        strcat(sub,bphone);
        strcat(sub," : ");
        strcat(sub,bdate);
        strcat(sub," : ");
        strcat(sub,broom);
        strcat(sub," : ");
        strcat(sub,btype);
        strcat(sub,"\n");

        fputs(sub,cuFile);

    }

}


fclose(cuFile);

if(check){

    printf("Can not check out ,can not find your log");

    return;

}


cuFile = fopen(strcat(type,".txt"), "r");

char n[25];

char s[25];

Room *r, *ar;

int first = 1;


while (fscanf(cuFile, "%s : %s", n,s) != EOF) {

    char *n_in = (char*)malloc(sizeof(n));

    strcpy(n_in,n);

    Room *in = (Room *) malloc(sizeof(Room));

    in->room_num = n_in;

    int st;

    if(strcmp(s,"Avalible") == 0)

        st = 1;

    else

        st = 0;

    in->status = st;
    in->next = NULL;

    if (first) {

        r = in;

        ar = r;

        first = 0;

    } else {

        r->next = in;

        r = r->next;

    }

}


fseek(cuFile, 0, SEEK_SET);

fclose(cuFile);


fopen(type, "w");

while (ar != NULL) {

    char buffer[255] = "";

    strcat(buffer, ar->room_num);

    strcat(buffer, " : ");


    if (strcmp(ar->room_num, droom) == 0) {

        strcat(buffer, "Avalible");

    }

    else {

        if (ar->status == 1)

            strcat(buffer, "Avalible");

        else

            strcat(buffer, "Full");
        }
        ar = ar->next;
        strcat(buffer, "\n");
        fputs(buffer, cuFile);
    }
    fclose(cuFile);
    printf("Check Out Success\n");
}



int check_duplicate(char* filename){
    FILE *file;
    file = fopen("Hotelname.txt","r");
    char buffer[strlen(filename)+1];
    while (fgets(buffer, sizeof(buffer),file)){
        if(strcmp(filename,buffer) == 0)
            return 0;
    }
    return 1;
}

void addHotel(){
    int firstdes =1;
    FILE *hotelfile,*room,*hotel;
    hotelfile = fopen("Hotelname.txt","a");

    char hotelname[255];
    printf("Enter Hotelname :");
```

```
scanf("%s", hotelname);
seek_to_next_line();
if(check_duplicate(hotelname)) {
   strcat(hotelname, "\n");
   fputs(hotelname, hotelfile);
   strtok(hotelname, "\n");
}
fclose(hotelfile);

char *hotel_room = (char*)malloc(sizeof(hotelname));
hotel_room = hotelname;
strcat(hotel_room,"_room.txt");
room = fopen(hotel_room,"a");
strtok(hotel_room,"_");

char moreRoom;

do{
   char roomtype[] = "Room Type : ";
   char buffer[255];
   char rt[255] = "";
   printf("Enter Room Type : ");
   scanf("%s",buffer);
   seek_to_next_line();

   strcat(buffer,".txt\n");
   strcat(rt,buffer);
   fputs(buffer,room);
```

```
strtok(buffer,".");
fclose(room);

strcat(hotelname,".txt");
hotel = fopen(hotelname,"a");
strcat(roomtype,buffer);
strcat(roomtype,"\n");
fputs(roomtype,hotel);

int check = 0;
do {
  char des[50];
  if(firstdes)
    strcpy(des,"    Description :  ");
  else
    strcpy(des,"                :  ");
  printf("Enter Description(1 line per 1 option) : ");
  scanf("%[^\n]", buffer);
  seek_to_next_line();

  strcat(buffer, "\n");
  strcat(des, buffer);
  fputs(des, hotel);

  char more;
  printf("Do you want to add more description? (y to add more):");
  scanf("%c",&more);
  seek_to_next_line();
```

```
    printf("You entered %c\n",more);


    if(more == 'y') {
        firstdes = 0;
        check = 1;
    }
    else {
        firstdes = 1;
        check = 0;
    }
}while (check == 1);


char people[] = "     People suggest: ";
char detail[255];
printf("Enter people suggest: ");
scanf("%s",detail);
seek_to_next_line();
strcat(detail,"\n");
strcat(people,detail);
fputs(people,room);


char price[] = "     Price: ";
char p[10];
printf("Enter the price :");
scanf("%s",p);
seek_to_next_line();


strcat(p,"\n");
strcat(price,p);
fputs(price,room);

int check_digit = 1;
do {
    char sroom[10], eroom[10];
    printf("Enter strating room number: ");
    scanf("%s", sroom);
    seek_to_next_line();

    for(int i =0;i < strlen(sroom);i++){
        if(!isdigit(sroom[i])){
            printf("Invalid Starting room\n");
            continue;
        }
    }

    printf("Enter ending room number: ");
    scanf("%s", eroom);
    seek_to_next_line();

    for(int i =0;i < strlen(eroom);i++){
        if(!isdigit(sroom[i])){
            printf("Invalid Ending room\n");
            continue;
        }
    }
    if (check_digit) {
        FILE *roomCheck;
        roomCheck = fopen(strtok(rt,"\n"), "w");
        int s = atoi(sroom),e = atoi(eroom);
        if(s < e){
            while (s <= e){
                char b[10];
                itoa(s,b,10);
                strcat(b," : Avalible\n");
                fputs(b,roomCheck);
                s++;
            }
        }
        else{
            check_digit = 0;
            printf("starting room is less than ending room\n");
        }
    }
    else{
        printf("Invalid room\n");
    }
}while (check_digit == 0);




printf("Do you want to add more room type(y to add): ");
scanf("%c",&moreRoom);
seek_to_next_line();
```

```
        printf("You entered %c\n",moreRoom);
    }while (moreRoom == 'y');


    fclose(hotelfile);
    fclose(room);
    fclose(hotel);
}


void menuAdmin(){
    int command , check = 0;
    do {
        printf("1.Adding Hotel\n");
        printf("2.Check out\n");
        printf("3.Back\n");
        printf("Enter the number of command :");
        scanf("%d", &command);
        if(command == 1)
            addHotel();
        else if(command == 2) {
            char room[10],name[25];
            printf("Enter room to check out: ");
            scanf("%s",room);
            seek_to_next_line();


            printf("Enter customer name:");
            scanf("%s",name);
            seek_to_next_line();
            checkout(room,name);
        }
        else if(command == 3)
            check = 1;
        else
            printf("Invalid command!!!");
    }while(check == 0);

}

void login(){
    char user[10],pass[10];
    int check = 0;
    do {
        printf("Enter Username (b to back): ");
        scanf("%s", user);
        if (strcmp(user, "b") == 0)
            return;
        printf("Enter Password: ");
        scanf("%s", pass);
        int check_user = strcmp(user, "Admin"),
                check_pass = strcmp(pass, "rootAdmin");
        if (check_user && check_pass) {
            printf("Username or Password is invalid\n");

        } else {
            menuAdmin();
            check = 1;
        }
    }while(check == 0);
}

#endif //BOOKINGHOTEL_ADMIN_H
```

User.h

```
#ifndef BOOKINGHOTEL_USER_H

#define BOOKINGHOTEL_USER_H


#include <stdio.h>

#include <string.h>

#include <stdlib.h>


void information(char room_number[10],char room_type[50]){

    char name[128],number[128],date[128];


    printf("Enter your name:");

    scanf("%s",name);


    printf("Enter your phone number:");

    scanf("%s",number);


    printf("Enter your booking date:");

    scanf("%s",date);


    char inf[255] =  "";

    strcat(inf,name);

    strcat(inf," : ");

    strcat(inf,number);

    strcat(inf," : ");

    strcat(inf,date);

    strcat(inf," : ");

    strcat(inf,room_number);
    strcat(inf," : ");
    strcat(inf,strtok(room_type,"."));
    strcat(inf,"\n")
    FILE *file;
    file  = fopen("Customer.txt","a");
    fputs(inf,file);
    fclose(file);
}

void search(int version){
    FILE *file;
    file  = fopen("Customer.txt","r");
    char name[128],number[128],date[128],input[128],room[128],type[128];
    char c;

    do {
        if (version == 0) {
            printf("Enter name to search:");
            scanf("%s", input);
        } else if (version == 1) {
            printf("Enter phone number to search:");
            scanf("%s", input);

        int check = 0;
        while (fscanf(file, "%s : %s : %s : %s : %[^\n]", name, number, date, room, type) != EOF) {
            if (version == 0) {
                if (strcmp(name, input) == 0) {
                    printf("%s : %s : %s : %s : %s\n", name, number, date, room, type);
                    check = 1;

                }

            } else if (version == 1) {

                if (strcmp(number, input) == 0) {

                    printf("%s : %s : %s : %s : %s\n", name, number, date, room, type);

                    check = 1;

                }

            }

        }


        if (check == 0 && version == 0) {

            printf("Can not find this name !!!\n");

        }

        else if (check == 0 && version == 1){

            printf("Can not find this name !!!\n");

        }

        printf("Do you want to search again?(y to search or b to back)");

        getchar();

        scanf("%c", &c);

        if(c == 'b')

            return;

    }while(c == 'y');



}


#endif //BOOKINGHOTEL_USER_H
```