**床位分配**

#include<stdio.h>

#include<stdlib.h>

typedef struct hotel\_level //旅店等级节点

{

int level; //房间等级

struct hotel\_ \*next; //下一等级的指针

struct room \*r\_next; //该等级下一个房间的指针

} hotel;

typedef struct room //房间节点

{

int bednum; //床位数

int sex; //性别

int membernum; //入住人数

int roomnum; //房间号

struct room \*next; //下一个房间的指针

struct customer \*next\_c; //下一个旅客的指针

} room;

typedef struct customer //旅客节点

{

char name[8]; //姓名

int sex; //性别

int age; //年龄

int level; //等级

int room\_num; //房间号

int bed;

char date[20];

struct customer \*next; //下一个旅客信息

} customer;

void init(hotel \*L)

{

int i,j,k,l;

hotel \*p,\*q;

room \*p1,\*p2,\*p3;

printf("请输入房间分为几个等级: ");

scanf("%d",&j);

q=L; //等级设置了头结点

for(i=1; i<=j; i++) //创建等级-房间-床位数

{

p=(hotel \*)malloc(sizeof(hotel));//开辟等级结点

q->next=p;

p->level=i; //给房间等级赋值

printf("请输入%d等级的房间个数: ",p->level);

scanf("%d",&l); //输入该等级的房间数目

if(l==0)

continue;

else

{

p2=(room \*)malloc(sizeof(room));

p2->roomnum=1;

printf("请输入%d等级的房间号为1的房间床位个数的最大值: ",p->level);

scanf("%d",&(p2->bednum)); //输入房间1的床位数目

p2->membernum=0; //该房间内当前人数赋为零

p->r\_next=p2;

p3=p2;

for(k=2; k<=l; k++)

{

p1=(room \*)malloc(sizeof(room));

p3->next=p1;

p1->roomnum=k;

printf("请输入%d等级的房间号为%d的房间床位个数的最大值: ",p->level,k);

scanf("%d",&(p1->bednum)); //输入房间k的床位数目

p1->membernum=0; //该房间内当前人数赋为零

p3=p1;

}

p3->next=NULL; //该等级房间床位设置结束，房间链表尾指针赋为空

}

q=q->next; //等级指针指向下一个等级

}

q->next=NULL; //所有等级设置结束，等级链表尾指针域赋为空

}

int isfull(hotel \*L) //是否客满函数，满则返回1，有空位则返回0

{

hotel \*l;

l=L->next;

room \*r;

while(l!=NULL)

{

r=l->r\_next;

while(r!=NULL)

{

if(r->membernum!=r->bednum)//入住人数不等于床的数量，说明没有满

return 0;

else

r=r->next;

}

l=l->next;

}

if(!l)

return 1;

}

void printc(room \*q,customer \*p)

{

printf(" 旅客信息 \n");

printf("姓名:%s\n",p->name);

printf("性别:%d\n",p->sex);

printf("年龄:%d\n",p->age);

printf("等级:%d\n",p->level);

printf("房间号:%d\n",p->room\_num);

printf("床位号:%d\n",p->bed);

printf("入住时间:%s\n",p->date);

printf("\n");

}

void input(hotel \*L)

{

char c;

customer \*s,\*p2,\*p3;

hotel \*l;

l=L->next;

room \*p1;

int i,lv;

s=(customer \*)malloc(sizeof(customer));

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

printf("\n 旅客信息: ");

printf("\n请输入旅客姓名:");

scanf("%s",s->name);

printf("\n 请输入旅客年龄:");

scanf("%d",&(s->age));

printf("\n 请输入旅客性别(男1 女0):");

scanf("%d",&(s->sex));

printf("\n 请输入住日期:");

scanf("%s",s->date);

if(isfull(L))

{

printf(" 旅店客满 ，无法入住。\n");

}

else

{

printf("请输入旅客所需的房间等级:"); //输入旅客所需的等级

scanf("%d",&lv);

while(l!=NULL) //等级不为空时

{

if(l->level!=lv)

l=l->next;

else //符合旅客需求的等级时

{

p1=l->r\_next;

while(p1!=NULL) //当房间存在时

{

if(p1->membernum==p1->bednum) //房间满时访问下一个房间

{

p1=p1->next;

if(p1==NULL)

{

getchar();

printf("此等级为%d的所有房间均无空位\n是否愿意更换等级[y/n]",lv);

scanf("%c",&c); //做出选择

if(c=='y')

input(L);

else

return ;

}

}

else //房间未满，进行旅客登记

{

//因为床位没有设置头结点，所以对于第一个结点要特别判断

if(p1->membernum==0)

{

s->level=lv;

p1->next\_c=s;

p1->membernum++; //房间当前人数加1

s->bed=1; // 顾客床位号

s->next=NULL; //尾指针置为空

printc(p1,s); //打印房间分配表

break;

}

else

{

p2=p1->next\_c;

p3=p2;

i=1;

while(i<=p1->membernum)

{

if(i==p3->bed)

{

p3=p3->next;

i++;

}

else

{

i++;

break;

}

}

if(p3==NULL)//如果p3是最后一个结点（NULL的）时，输入旅客信息

{

s->level=lv;

while(p2->next!=p3)

p2=p2->next;

if(s->sex==p2->sex)

{

p2->next=s;

p1->membernum++;

s->bed=i;

s->next=NULL;

printc(p1,s);

break;

}

else p1=p1->next;

}

else

{

while(p2->bed==i)

p2=p2->next;

if(s->sex==p2->sex)

{

p2->next=s;

p1->membernum++;

s->bed=i;

s->next=NULL;

printc(p1,s);

break;

}

else p1=p1->next;

}

}

}

}

break;

}

}

}

}

customer \*delete(hotel \*L)

{

int lv,hotel\_num,bed\_num;

hotel \*l;

l=L->next;

room \*r;

customer \*p,\*q;

char c;

printf("请输入该退房人房间的等级号码：\n");

scanf("%d",&lv); //输入退房人的房间等级

getchar();

while(l!=NULL)

{

if(l->level!=lv)

l=l->next;

else //找到该等级

{

r=l->r\_next;

printf("请输入该退房人房间的房间号码：\n");

scanf("%d",&hotel\_num); //输入退房人的房间号码

getchar();

while(r!=NULL)

{

if(r->roomnum!=hotel\_num)

r=r->next;

else //找到该房间

{

p=r->next\_c;

q=r->next\_c;

printf("请输入该退房人的床位号码：\n");

scanf("%d",&bed\_num); //输入退房人的床位号码

getchar();

while(q!=NULL)

{

if(q->bed!=bed\_num)

q=q->next;

else //查找该床位

{

if(bed\_num==1) //因为床位没有设置头结点，所以对于第一个结点要特别判断

{

//直接打印旅客信息

printf("此等级、此房间、此床位旅客的信息为：\n");

printf("该名顾客的信息:\n");

printf("姓名\t性别\t年龄\t入住时间\n");

printf("%s\t%d\t%d\t%s\n",q->name,q->sex,q->age,q->date);

printf("是否与要退房人信息相符？[y/n]\n");

scanf("%c",&c);

if(c=='y') //做出是否退房的选择

{

r->membernum--;

printf("退房成功！\n");

return(q);

}

else

return NULL;

}

else //如果床位不为1

{

while(p->next!=q) //指向下一床位 通过循环找到该床位

p=p->next; //打印旅客信息

printf("此等级、此房间、此床位旅客的信息为：\n");

printf("该名顾客的信息:\n");

printf("姓名\t性别\t年龄\t入住时间\n");

printf("%s\t%d\t%d\t%s\n",q->name,q->sex,q->age,q->date);

printf("是否与要退房人信息相符？[y/n]\n");

scanf("%c",&c);

if(c=='y') //做出是否退房的选择

{

p->next=q->next;

r->membernum--;

printf("退房成功！\n");

return(q);

}

else

return NULL;

}

}

}

if(q==NULL) //如果该房间内床位为空

{

printf("未找到该床位号，请核实信息后重新输入。\n");

}

}

}

if(r==NULL) //如果该等级的房间为空

{

printf("未找到该房间号，请核实信息后重新输入。\n");

}

}

}

if(l==NULL) //如果等级链表为空

{

printf("未找到该房间等级，请核实信息后重新输入。\n");

}

}

void cx\_customer(hotel \*L) //查询旅客信息函数

{

char s[15];

hotel \*l;

room \*r;

customer \*t;

int flag=0; //设置标志量

printf("请输入要查询的旅客姓名：\n");

scanf("%s",s); //输入要查找旅客的姓名

getchar();

l=L->next;

while(l!=NULL) //等级不为空时，走向房间

{

r=l->r\_next;

while(r!=NULL) //房间不为空时，走向床位

{

if(r->membernum==0) //若此房间当前人数为零，走向下一房间

r=r->next;

else

{ t=r->next\_c;

while(t!=NULL)

{

if(!strcmp(t->name,s)) //如果找到该旅客

{

printc(r,t); //打印信息

t=t->next; //指向下一人，继续查找

flag++; //标志量加1

}

else

t=t->next;

}

r=r->next;

}

}

l=l->next;

}

if(!flag) //标志量为零，则未查找到

printf("未找到该旅客，请核实后再输入。\n");

}

void Tongji\_cus(hotel \*L) //统计旅店当前住宿人数函数

{

hotel \*l;

room \*r;

customer \*t;

int i=0,j=0;

int flag;

l=L->next;

while(l!=NULL)

{

flag=0;

i++;

r=l->r\_next;

while(r!=NULL)

{

if(r->membernum==0)

r=r->next;

else

{

t=r->next\_c;

while(t!=NULL)

{

flag++;

j=j+flag;

t=t->next;

}

r=r->next;

}

}

printf("等级为%d的房间入住人数为%d\n",i,flag);

l=l->next;

}

if(!j)

printf("此时无人住宿。\n");

}

int main()

{

hotel \*L;

customer \*p;

int c;

L=(hotel\*)malloc(sizeof(hotel));

printf("初始化旅店信息：\n");

init(L);

while(1)

{ //进入菜单选项

printf("\n\n\n");

printf(" |------------------------------------------------------------|\n");

printf(" | |\n");

printf(" | 欢迎登陆住宿管理系统 |\n");

printf(" | |\n");

printf(" |------------------------------------------------------------|\n");

printf(" 1. 录入旅客信息\n");

printf(" 2. 旅客退房 \n");

printf(" 3. 查询旅客信息\n");

printf(" 4. 统计旅客信息\n");

printf(" 5. 退出本程序 \n");

printf(" 请选择1-5： \n");

scanf("%d",&c); //做选择

getchar();

switch(c)

{

case 1: input(L);

break;

case 2: p=delete(L);free(p);

break;

case 3: cx\_customer(L);

break;

case 4: Tongji\_cus(L);

break;

case 5: return;

}

}

return 0;

}







