# 题目理解

# 理论分析

# 软件结构

# 函数介绍与流程图

由于本程序主函数简练，高度函数化，因此函数功能和结构略微复杂，现将函数等相关信息用表格列举如下，再做详细介绍：

|  |  |
| --- | --- |
| 函数原型 | 调用的函数 |
| int main(int argc, char \*argv[]) | read(), interface(), meal\_ragi(),  meal\_canc(),meal\_inqu(), meal\_chan() |
| void interface() |  |
| void print(Info guest) |  |
| int compare(Info guest[]) | print() |
| void dele\_gue(Info guest[],int guest\_number) |  |
| int read(Info guest[]) | dele\_gue(), read() |
| void meal\_regi(Info guest[],int guest\_number) |  |
| void meal\_canc\_option(Info guest[],int i) | interface(), meal\_canc(), dele\_gue() |
| void meal\_canc(Info guest[]) | meal\_canc\_option(), compare() |
| void meal\_inqu(Info guest[]) | print() |
| void meal\_rewr(Info guest[]) |  |
| void meal\_chan(Info guest[]) | meal\_rewr(), compare(), meal\_chan() |

## int main(int argc, char \*argv[])

功能：调用其他函数，形成程序

结构：

## void interface()

## void print(Info guest)

## int compare(Info guest[])

## void dele\_gue(Info guest[],int guest\_number)

## int read(Info guest[])

## void meal\_regi(Info guest[],int guest\_number)

## void meal\_canc\_option(Info guest[],int i)

## void meal\_canc(Info guest[])

## void meal\_inqu(Info guest[])

## void meal\_rewr(Info guest[])

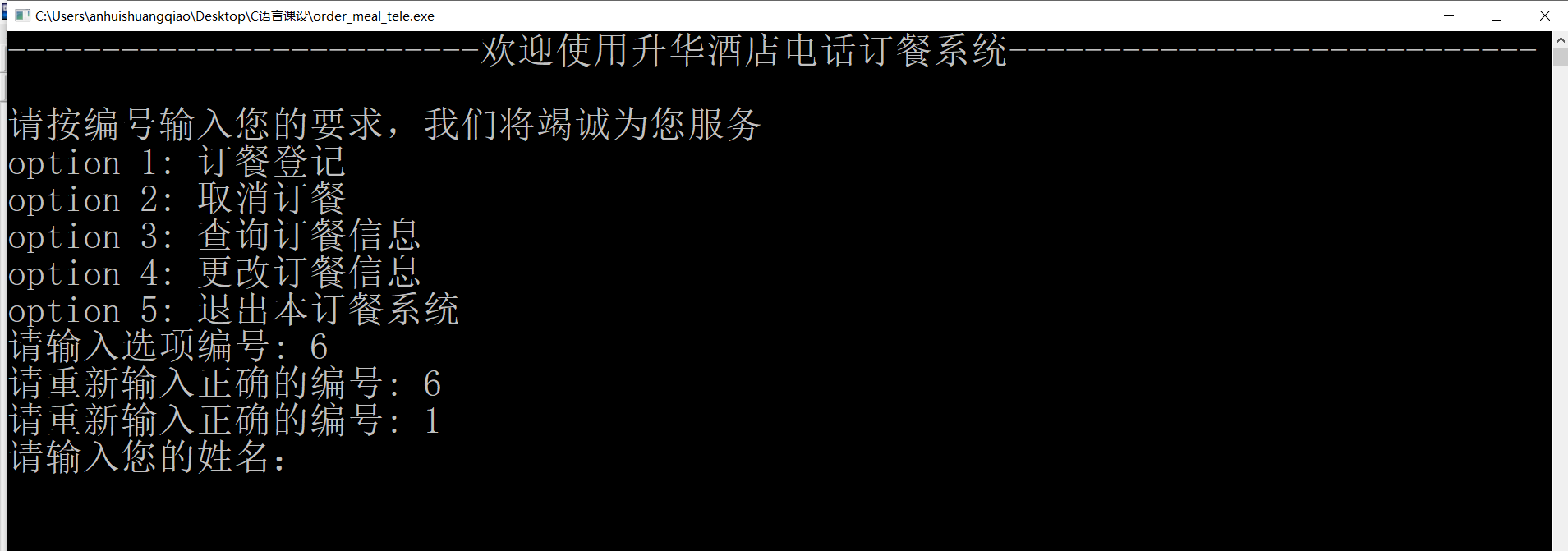
## void meal\_chan(Info guest[])

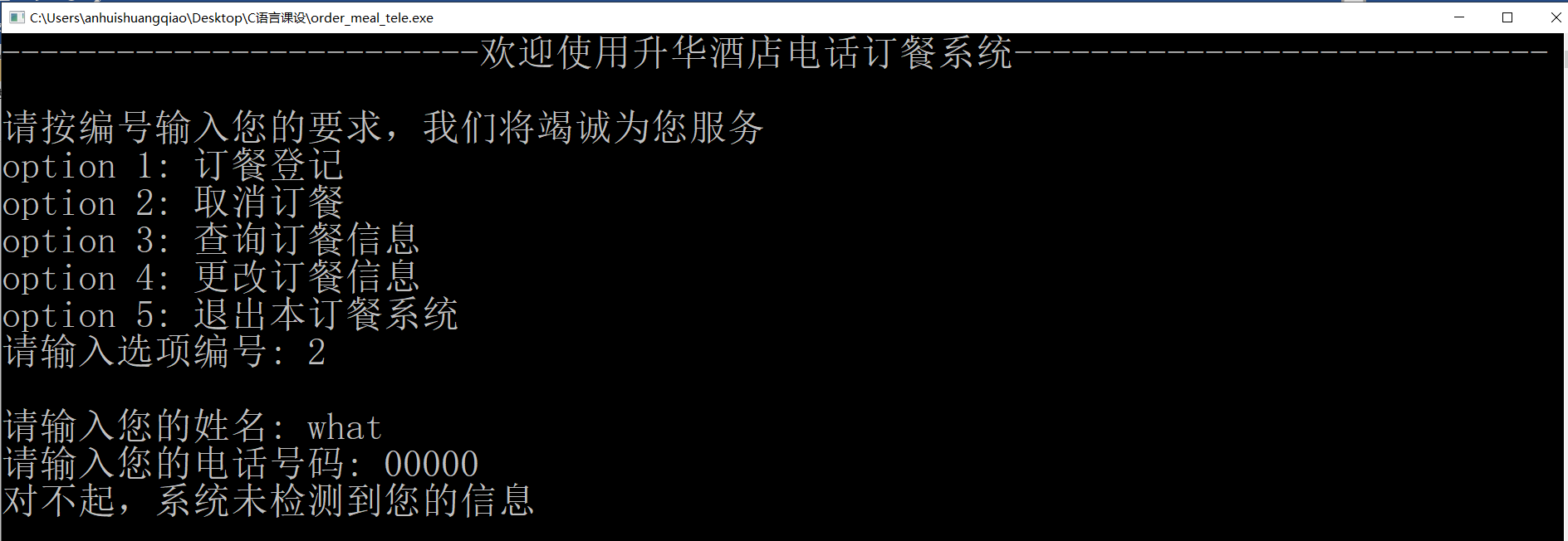
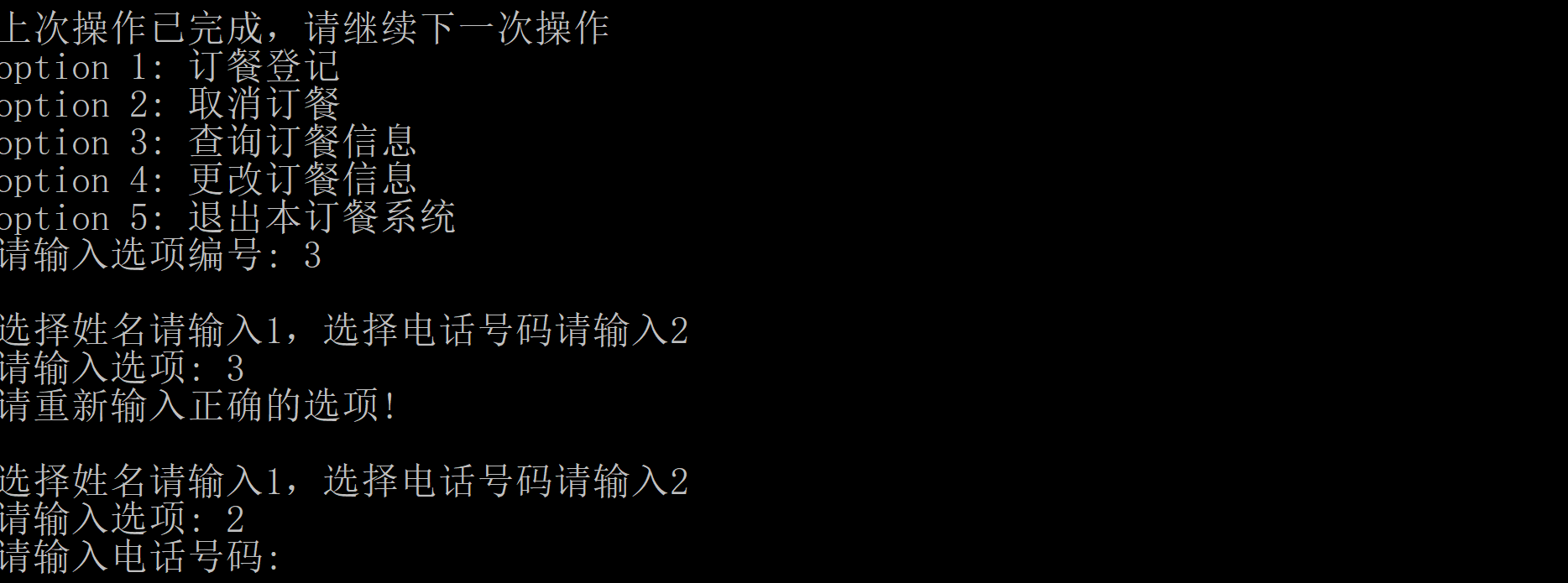
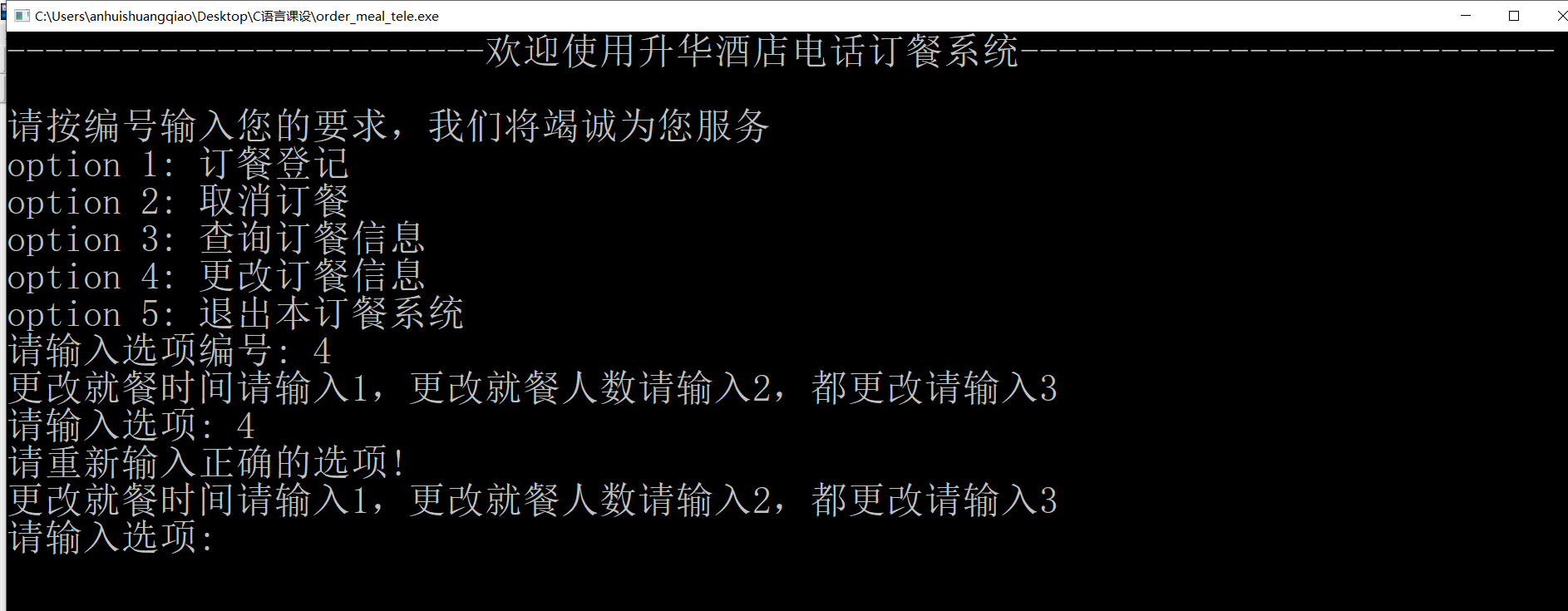
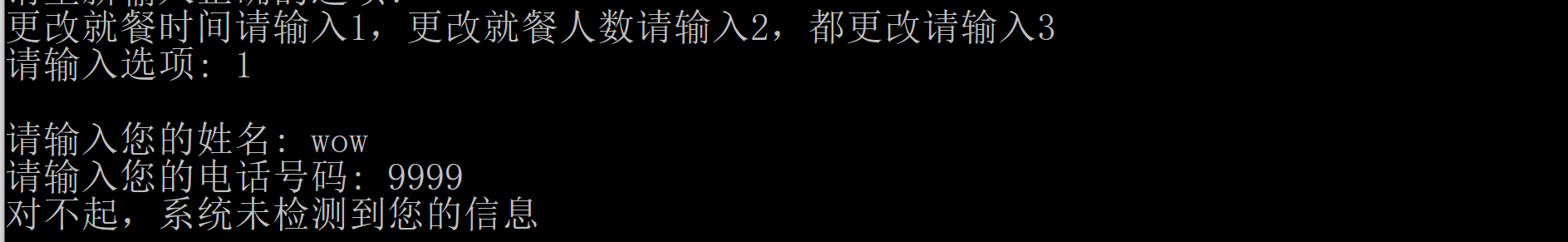
# 调试

# 亮点

## 防御性编程

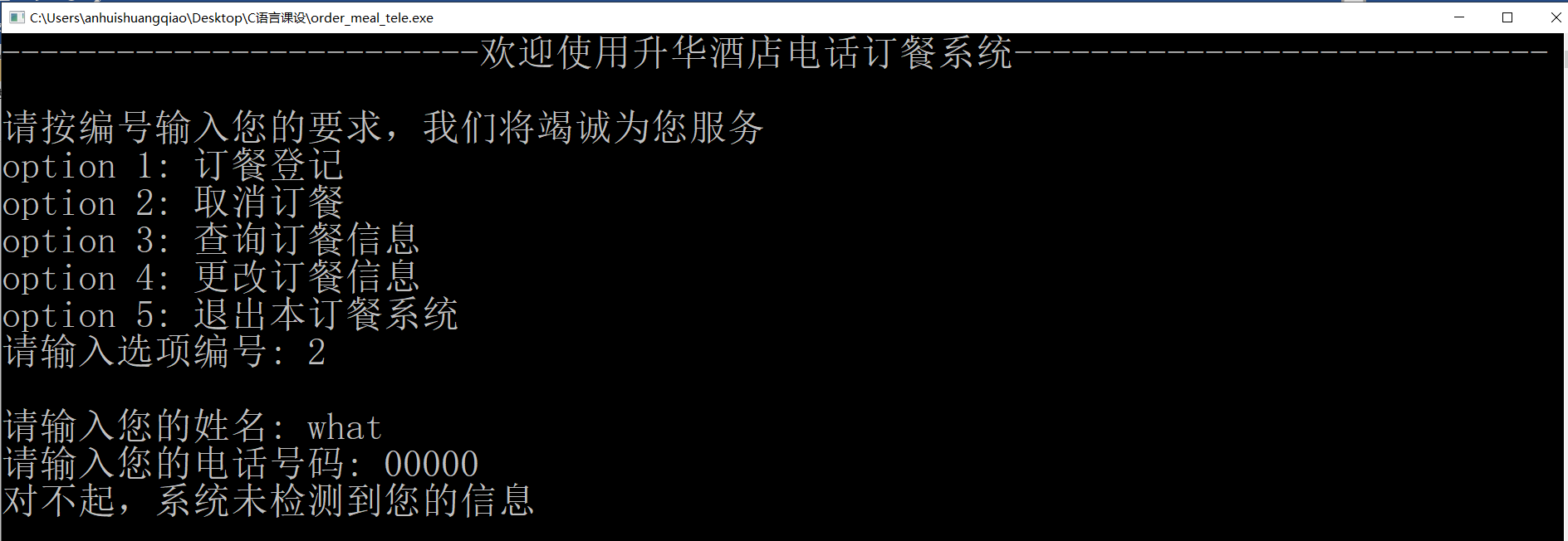
由于电话订餐系统的交互式特性，用户需要输入大量信息，假如选项输错可能会造成程序崩溃，因此，在编程中采用防御性思想，在用户给错信息的情况下给出错误提示并给出重新输入的机会。例如下列几幅图片：

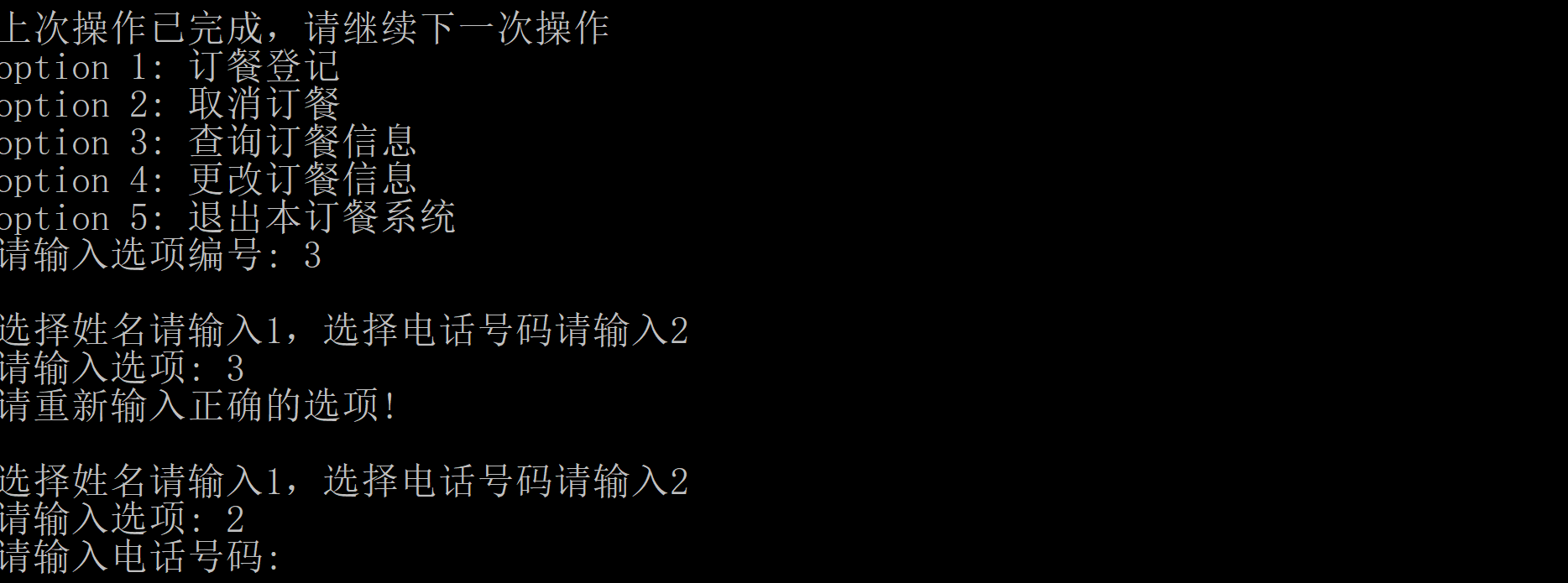


## 交互式界面，提示信息丰富且人性化

由于订餐系统面向用户，虽然控制台本身不够美观，但还是尽可能做出一个订餐系统应有的人机交互的感觉。例如以下几幅图片：



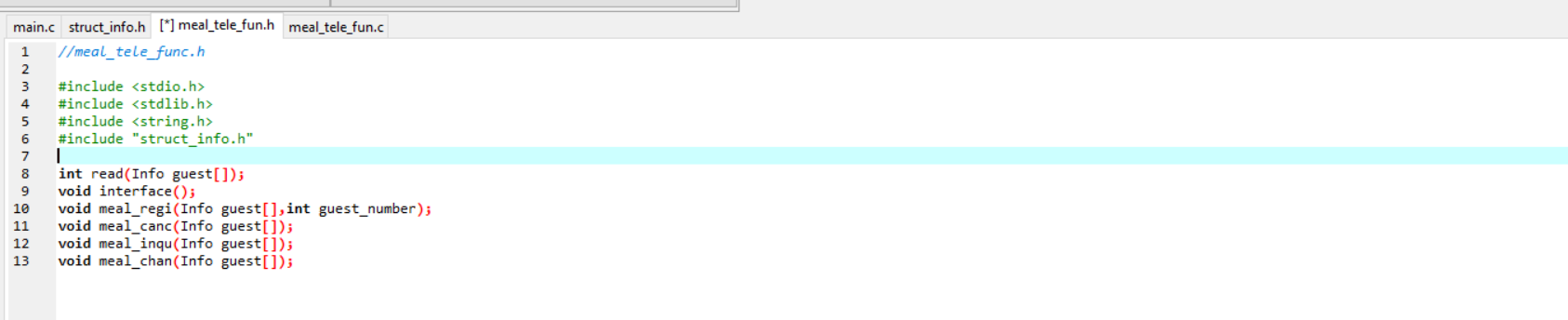


## 代码思路、结构清晰，高度函数化

为使总体架构清晰明了，在主函数main.c中仅写入主要的几个函数调用，使主函数仅达到35行。

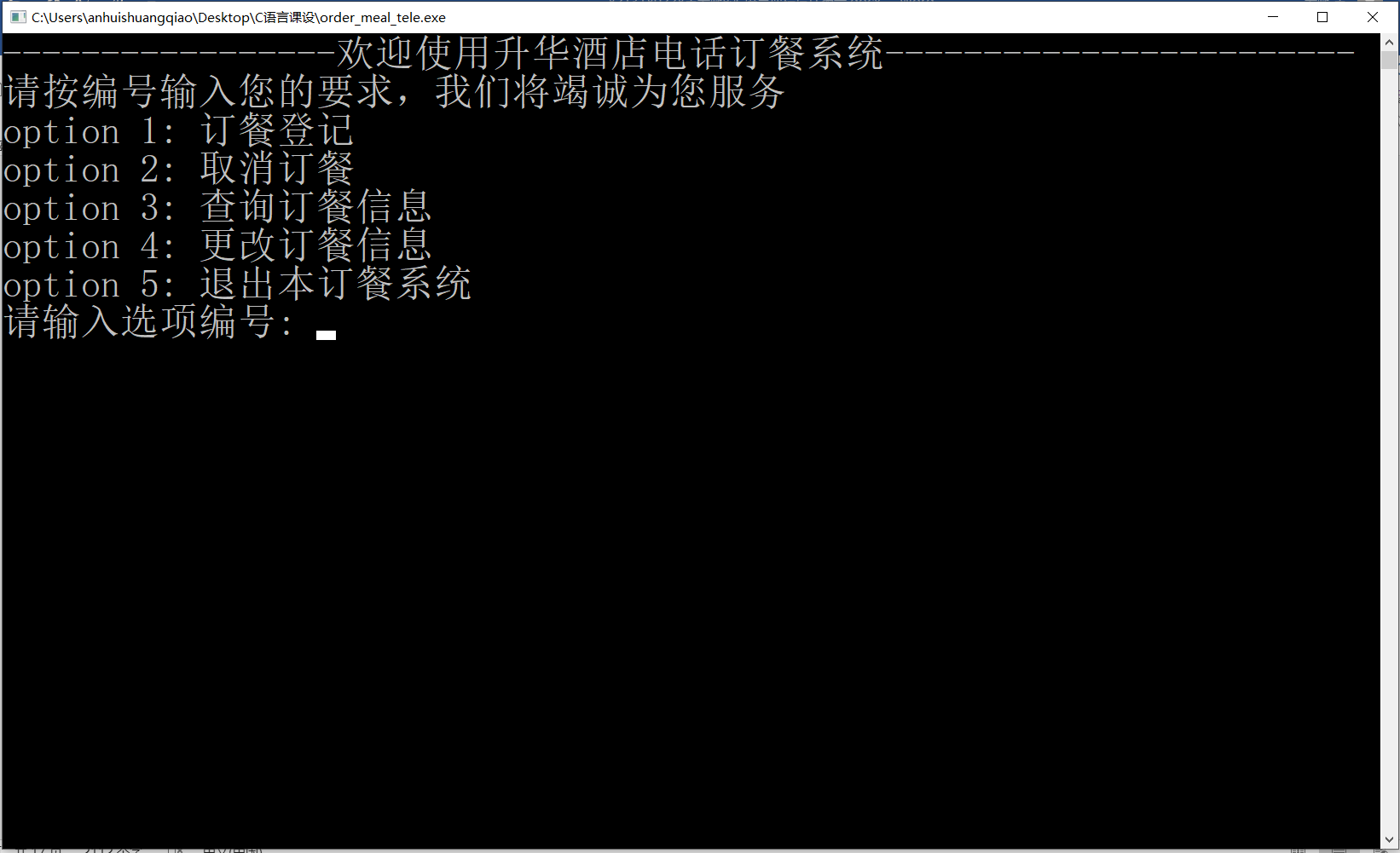


而将绝大部分代码写入另外的函数meal\_tele\_fun.c中，并将提供给主函数的函数声明写入meal\_tele\_func.h中，并令主函数包含之。



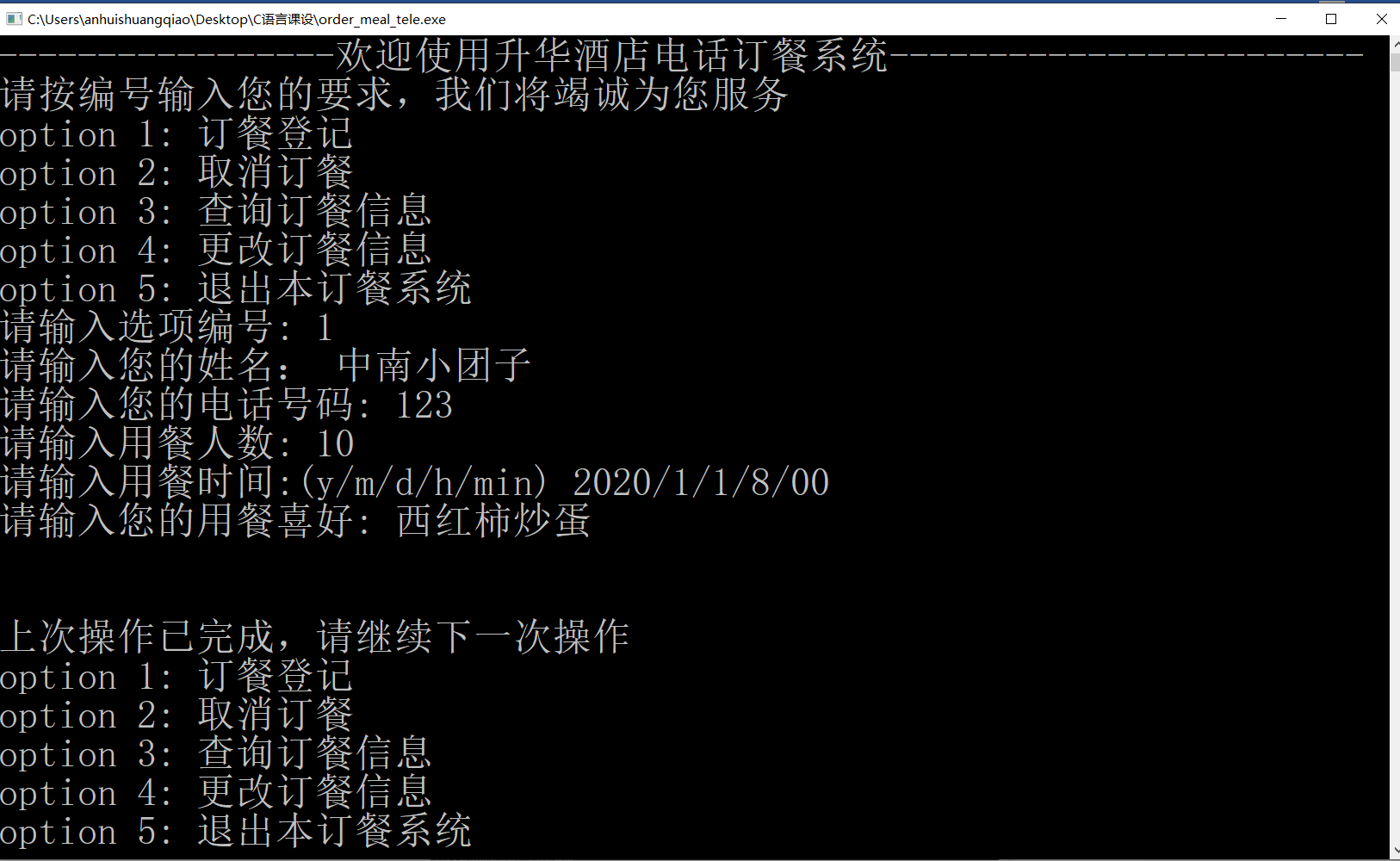
# 执行过程

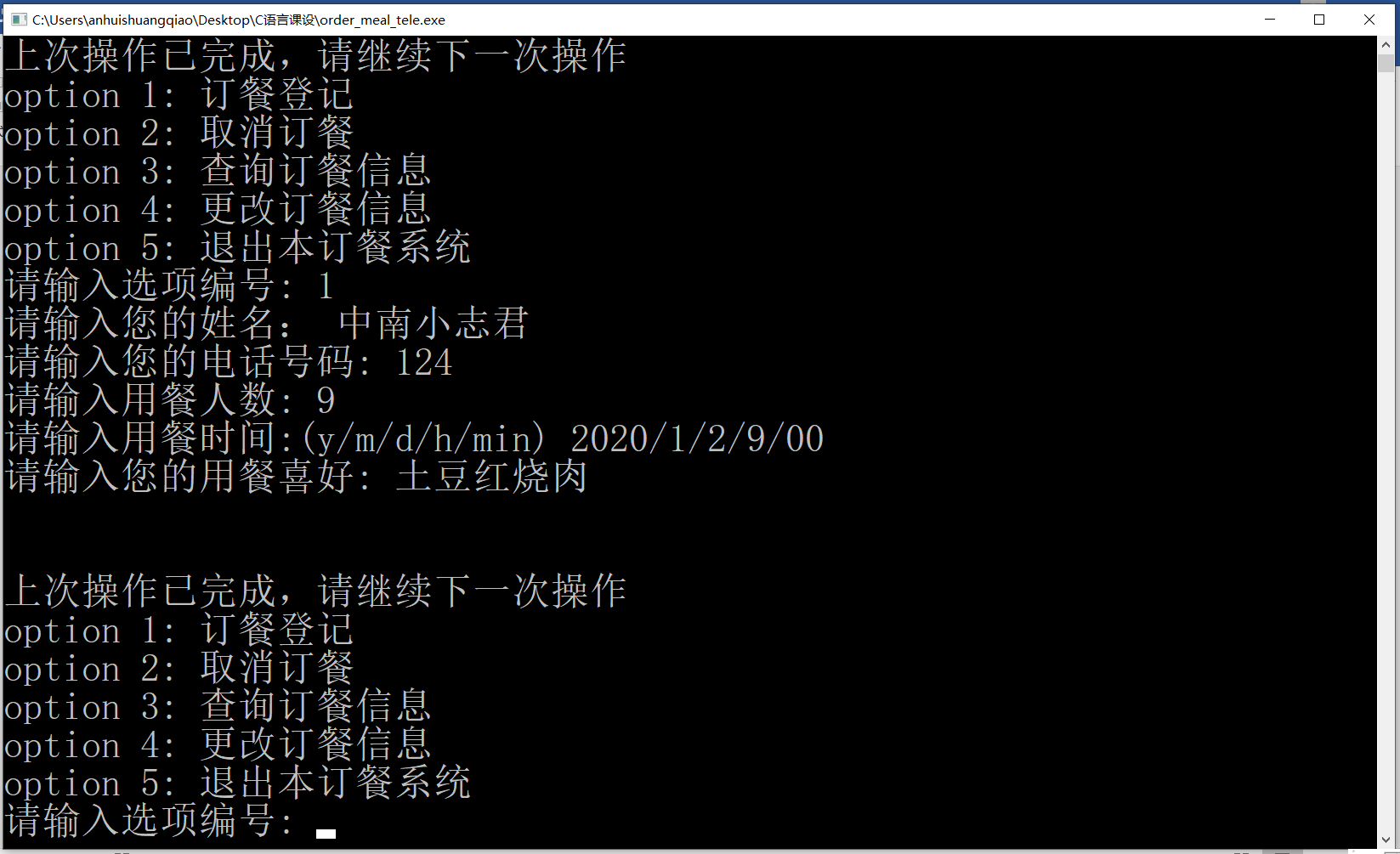
## 开始界面：



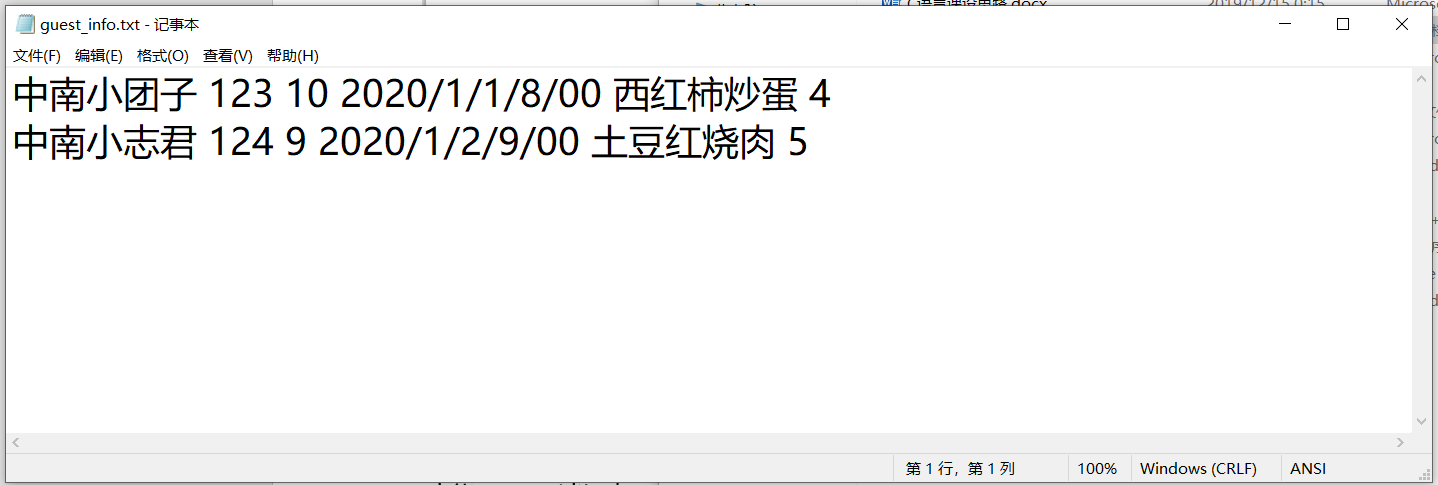
## 功能一：订餐登记

根据提示输入信息（下图为“中南小团子”、“中南小志君”分别订餐）





信息被存入TXT文件，在TXT文件中表示如下（订单编号即为每行最后一列数字）



## 功能二：取消订餐

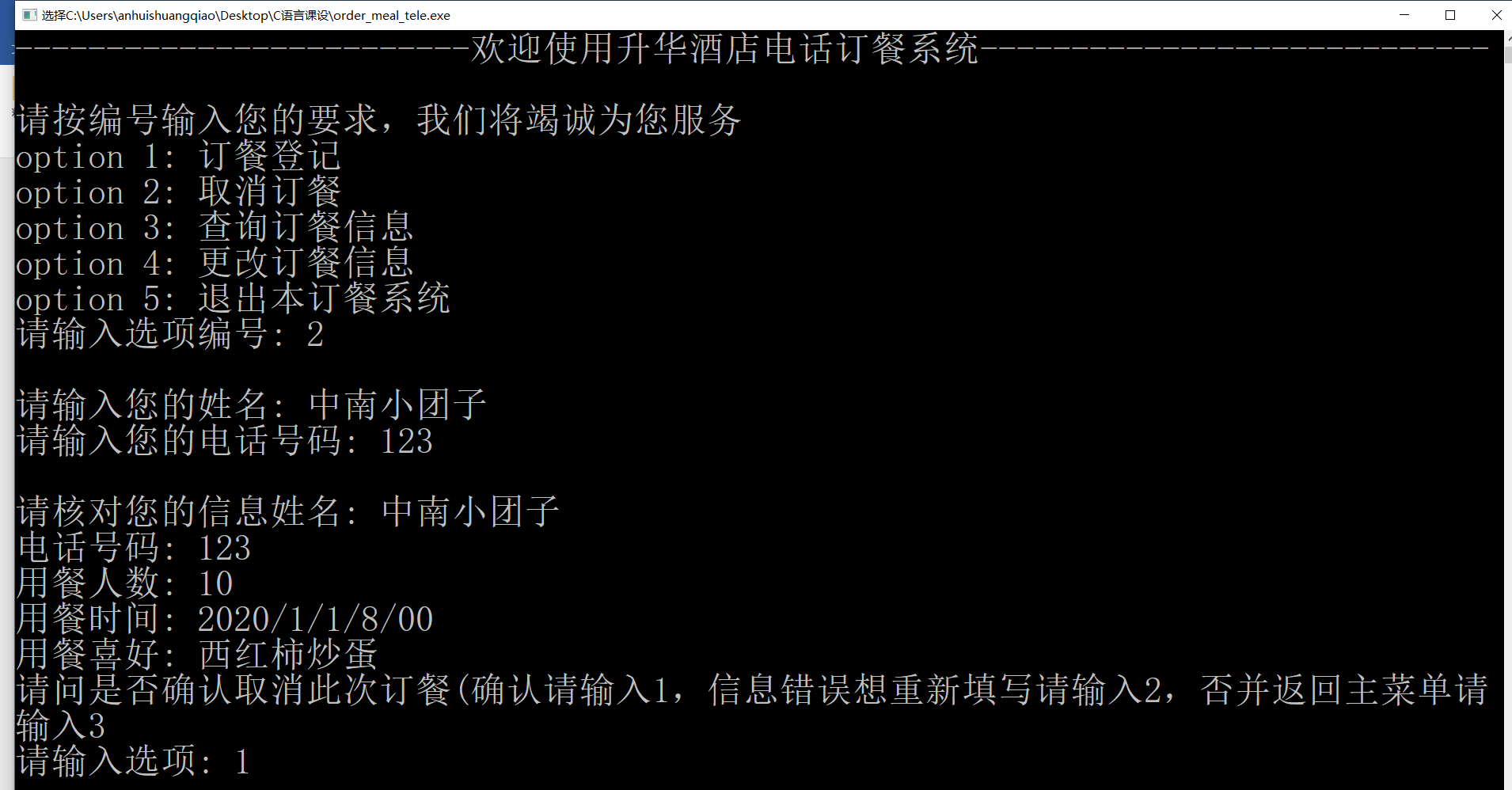
取消订餐有三个功能选项，分别为

1. 确认取消订餐
2. 拒绝取消订餐，返回主菜单

截图如下

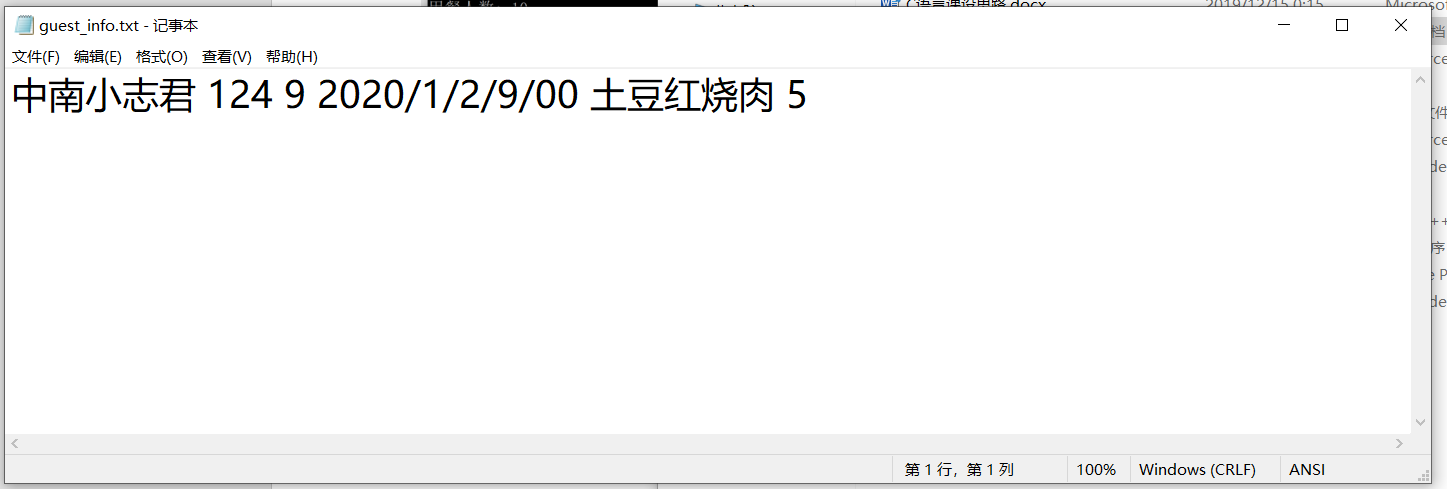
1. 确认取消订餐（下图为删除“中南小团子”的订餐）

检测到用户信息，提示是否确认

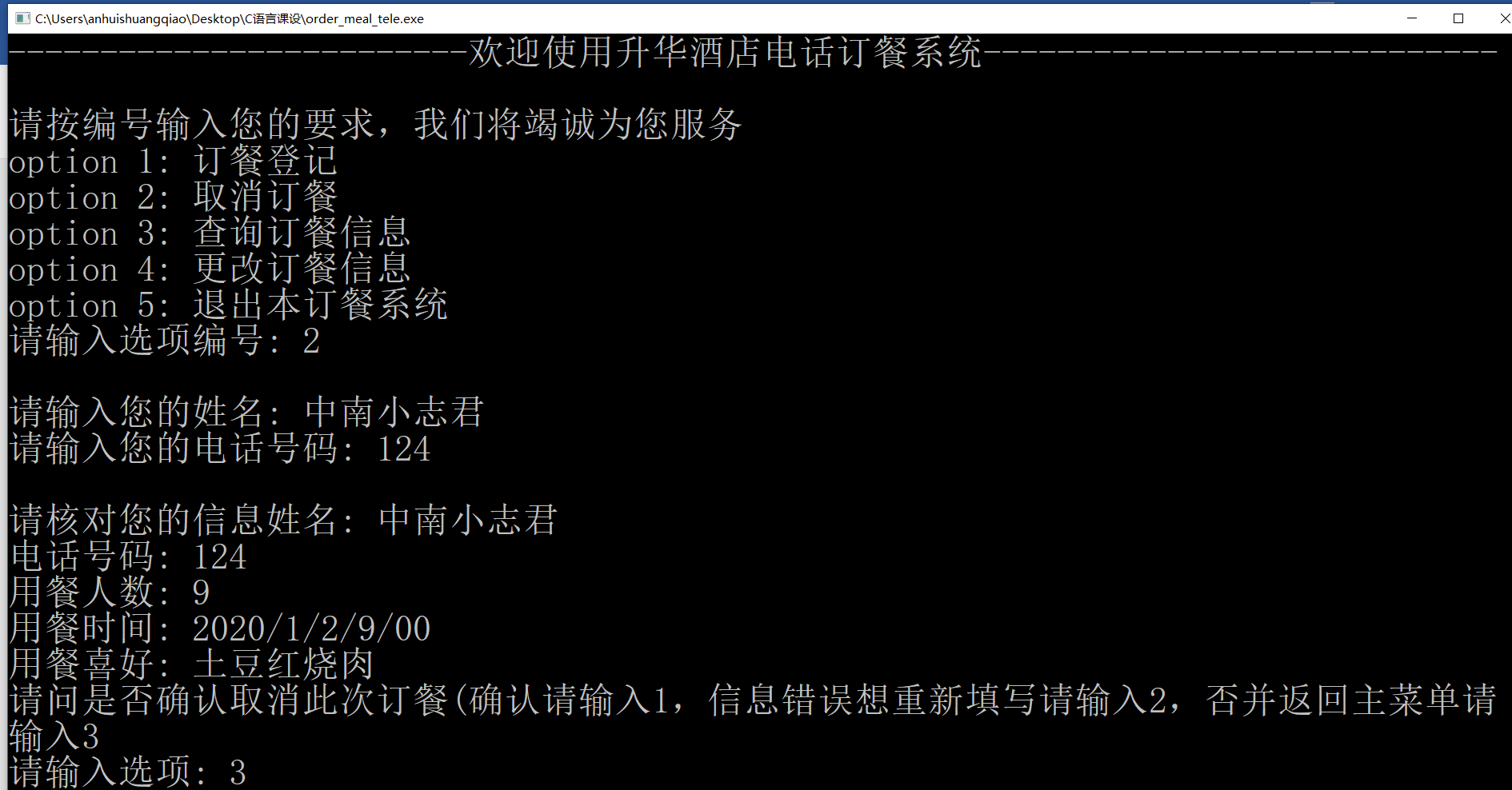


确认删除后，在TXT文件中表现如下

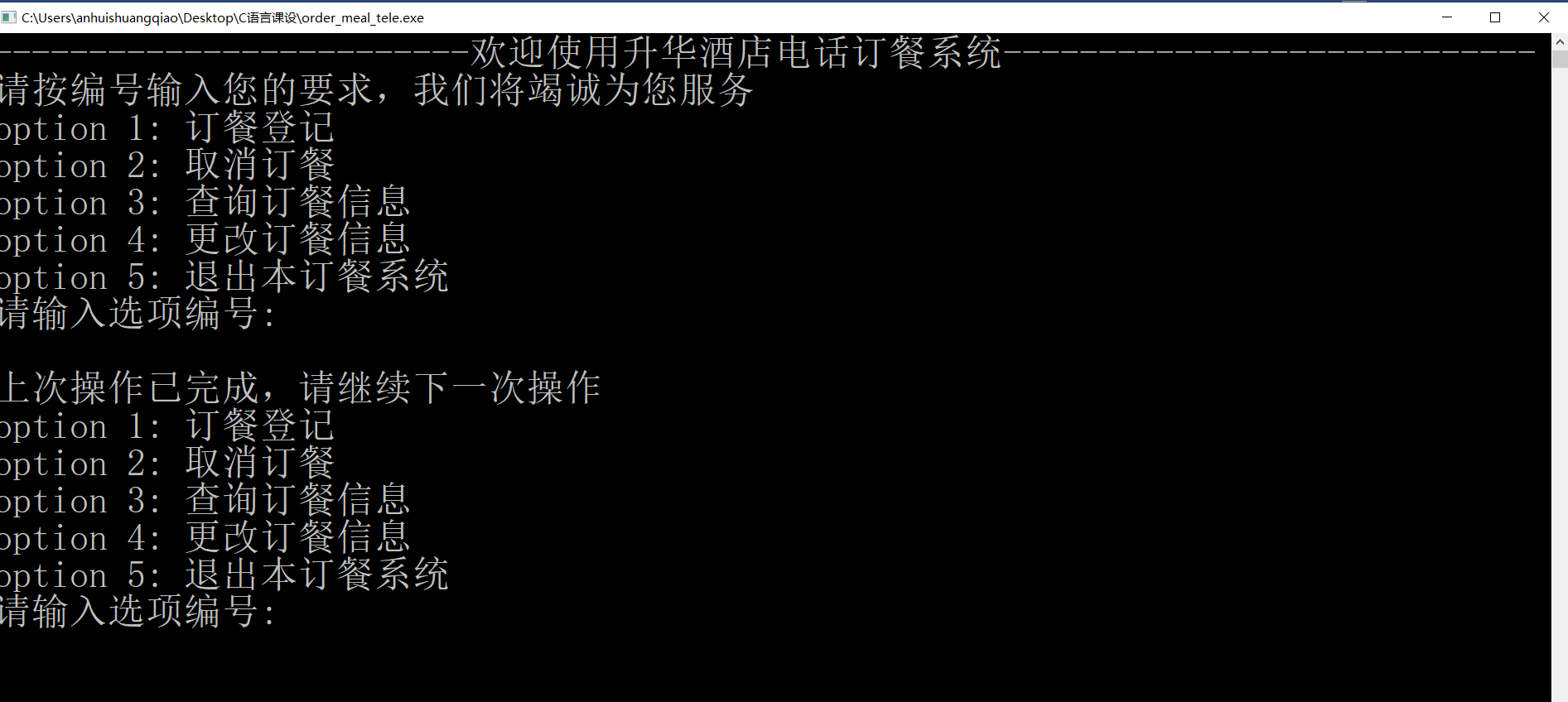
可以看到，中南小团子的订餐信息已被删除



1. 拒绝取消订餐，返回主菜单（用户输入3，拒绝取消，返回主菜单）



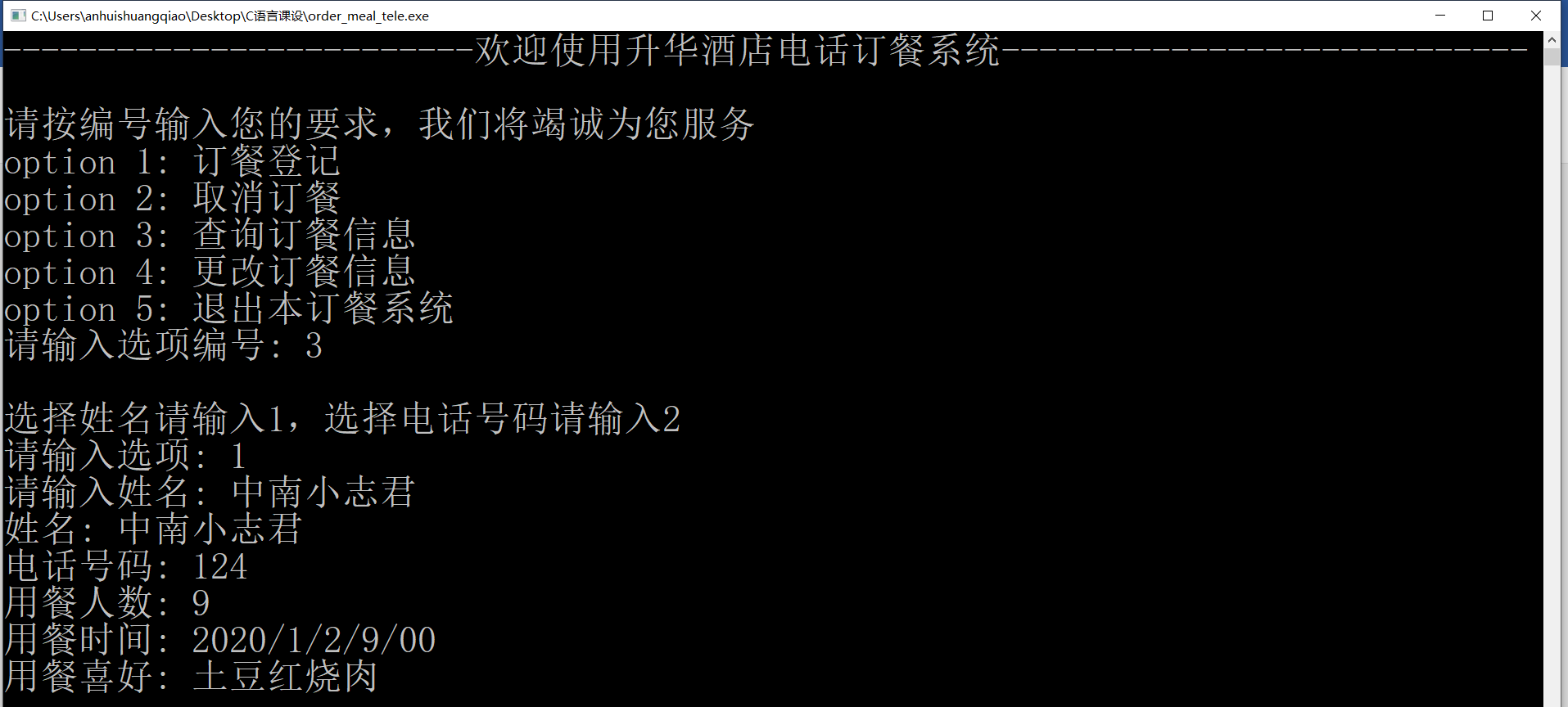
按下回车后



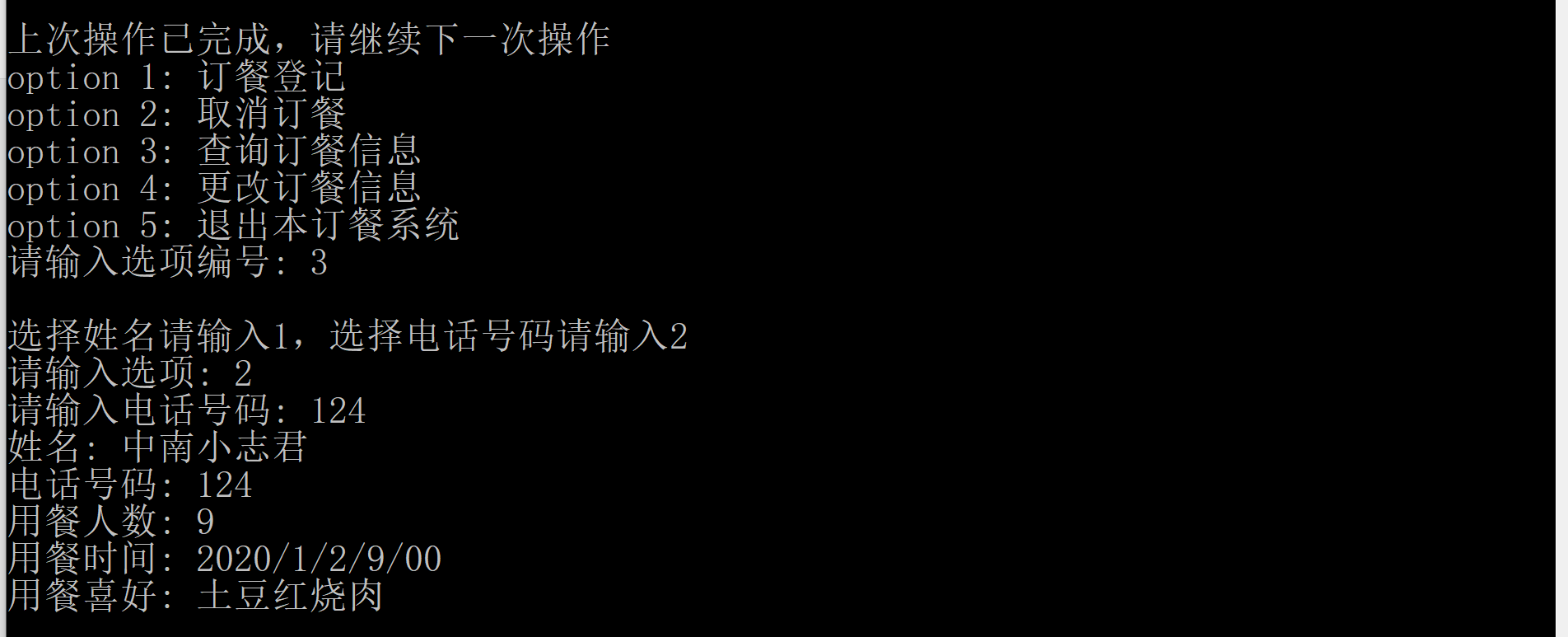
## 功能三：查询订餐信息

既可根据姓名查询，又可根据电话查询

1. 根据姓名查询（下图为用户输入选项1后，根据姓名查询“中南小志君”的订餐）



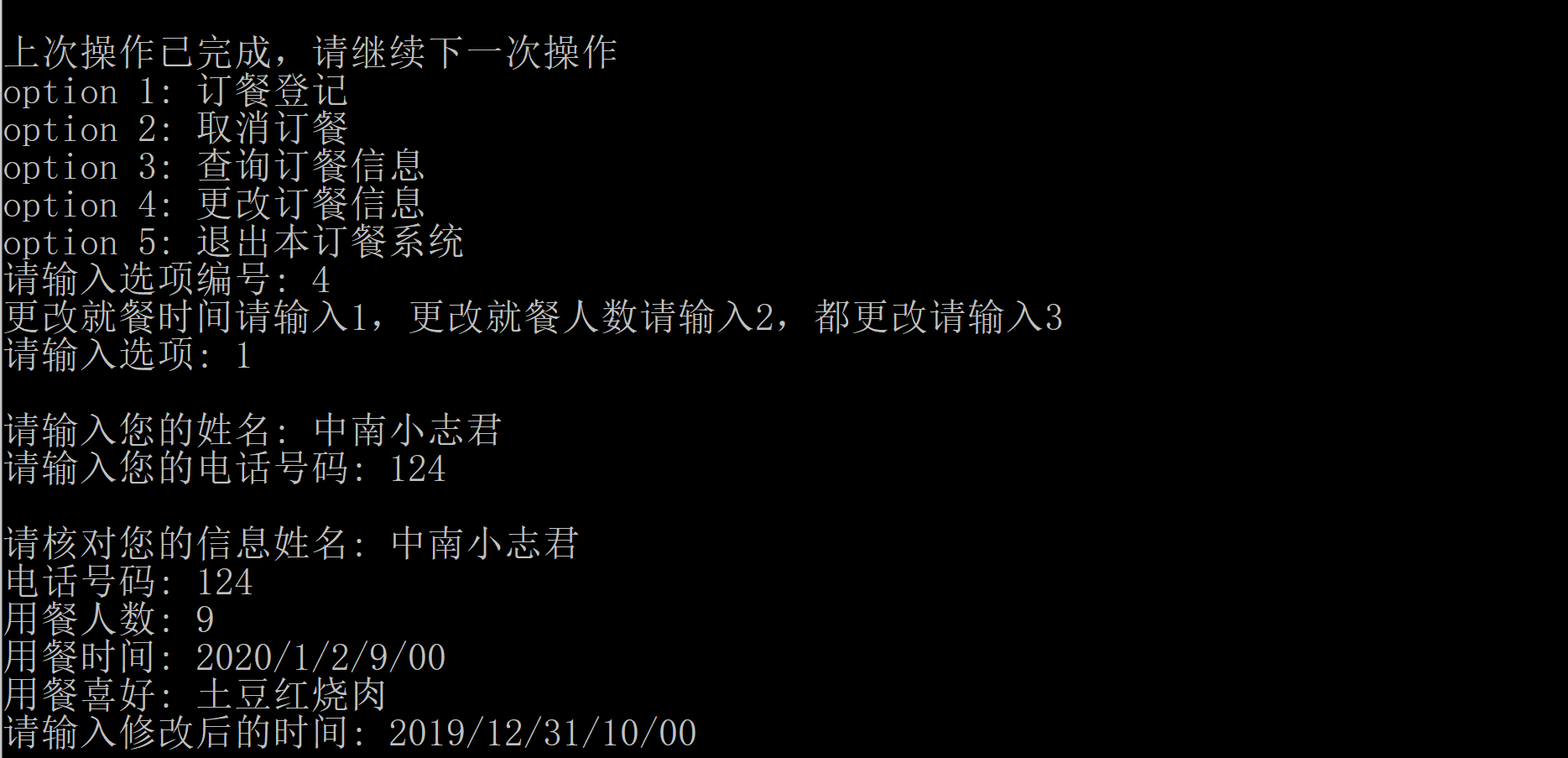
1. 根据电话查询（下图为用户输入选项二后，根据电话查询电话124的订餐）



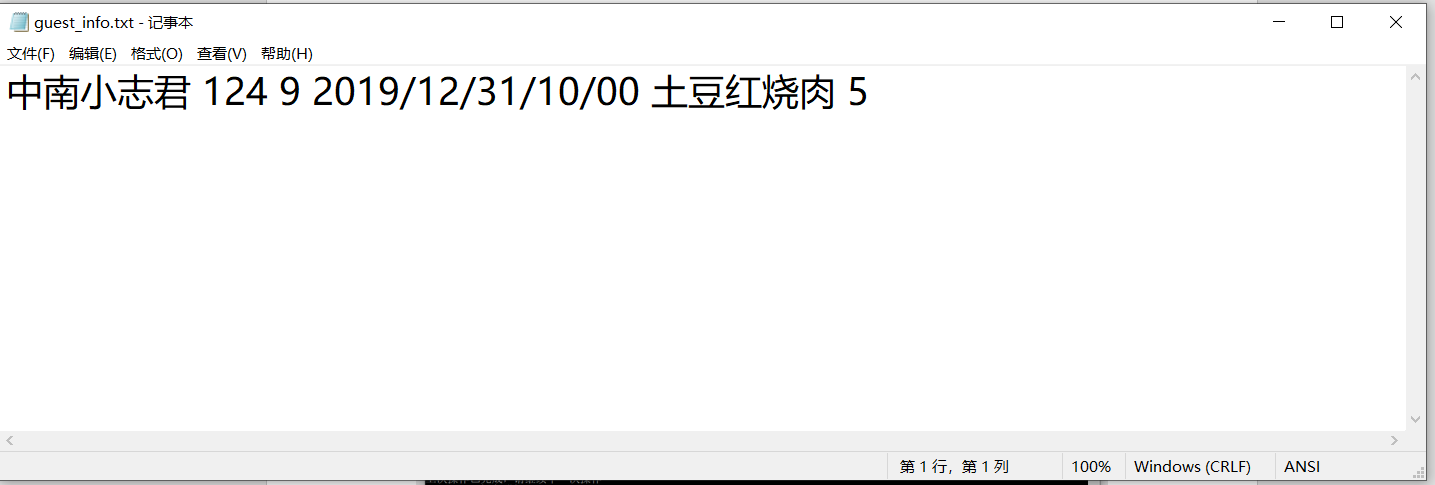
## 功能四：更改订单信息

可更改订餐时间，或更改用餐人数，或两者均更改

1. 更改订餐时间（下图为用户输入选项1后，更改“中南小志君”的订餐时间）

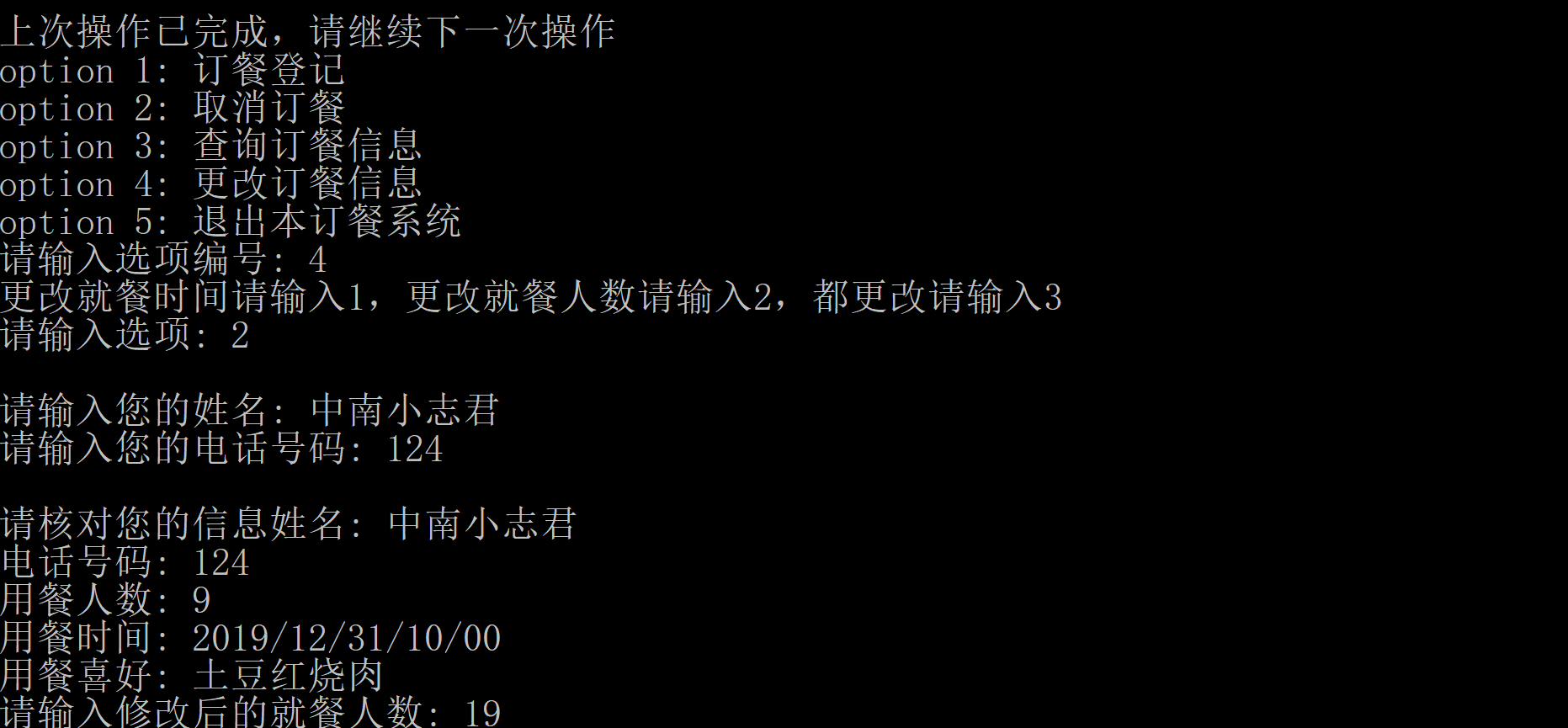


在TXT文件中表示如下



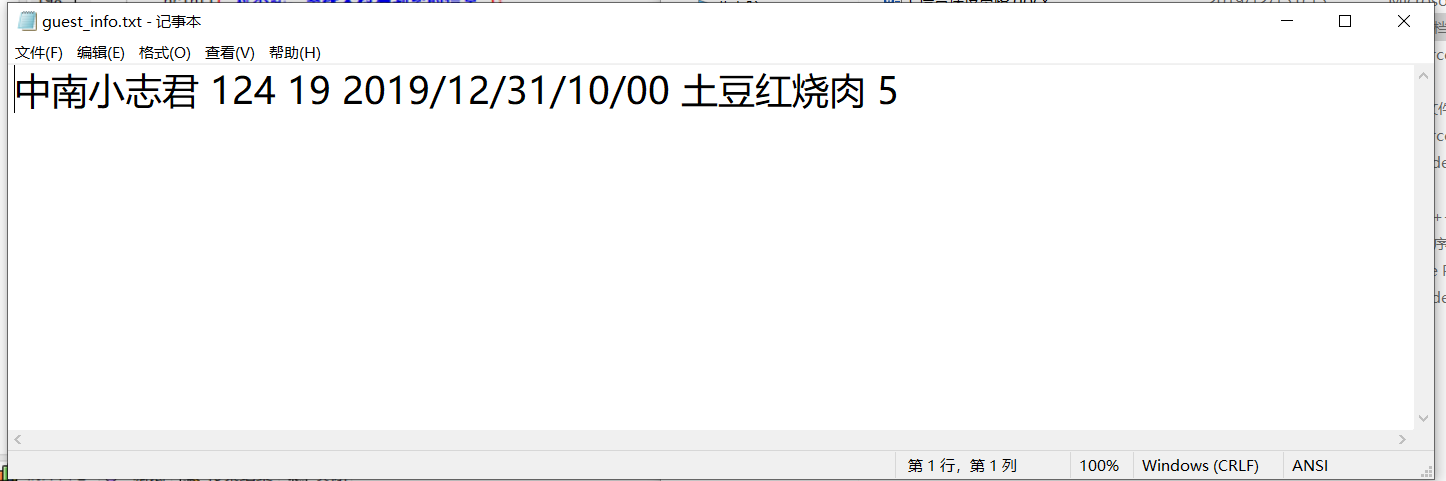
可以看到，中南小志君的订餐时间由2020/1/2/9/00修改为2019/12/31/10/00

1. 更改用餐人数（下图为用户输入选项二后，更改“中南小志”的用餐人数）



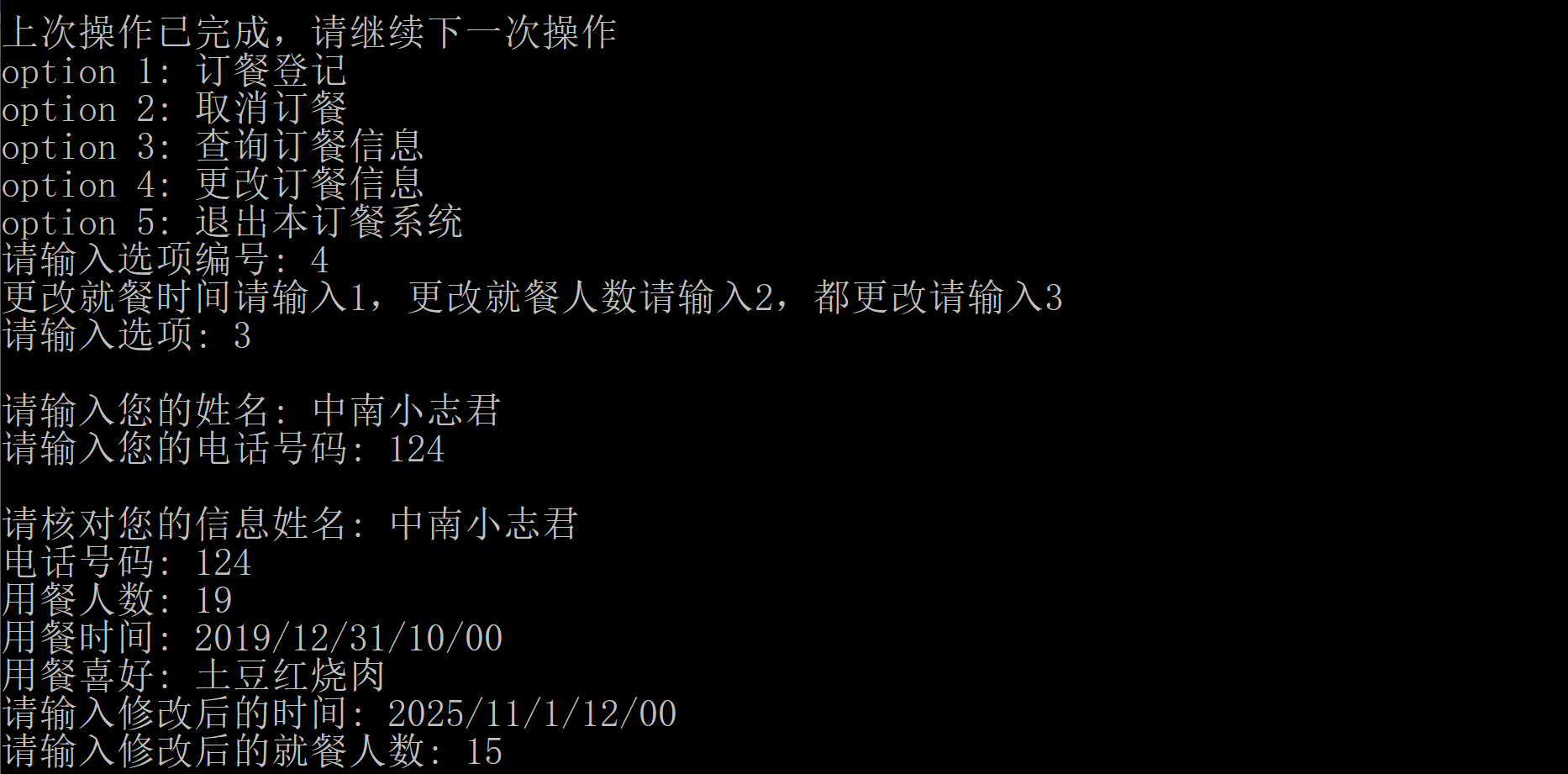
在TXT文件中表示如下

可以看到，中南小志君的用餐人数已从9变为19

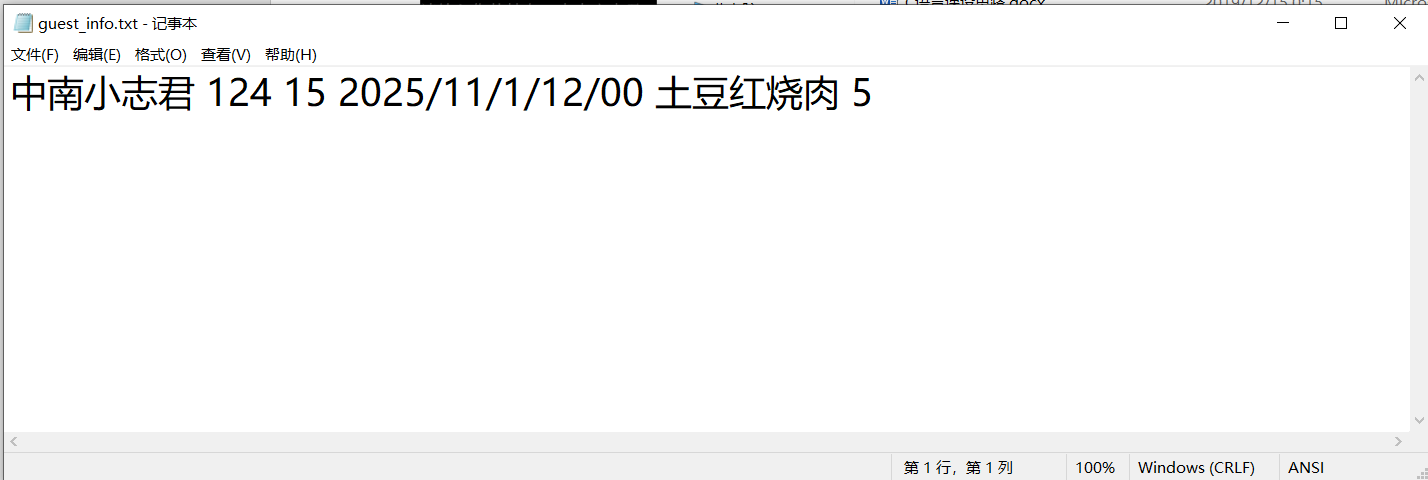


1. 更改订餐时间和用餐人数

下图为用户输入选项三后，修改”中南小志君“的订餐时间和用餐人数



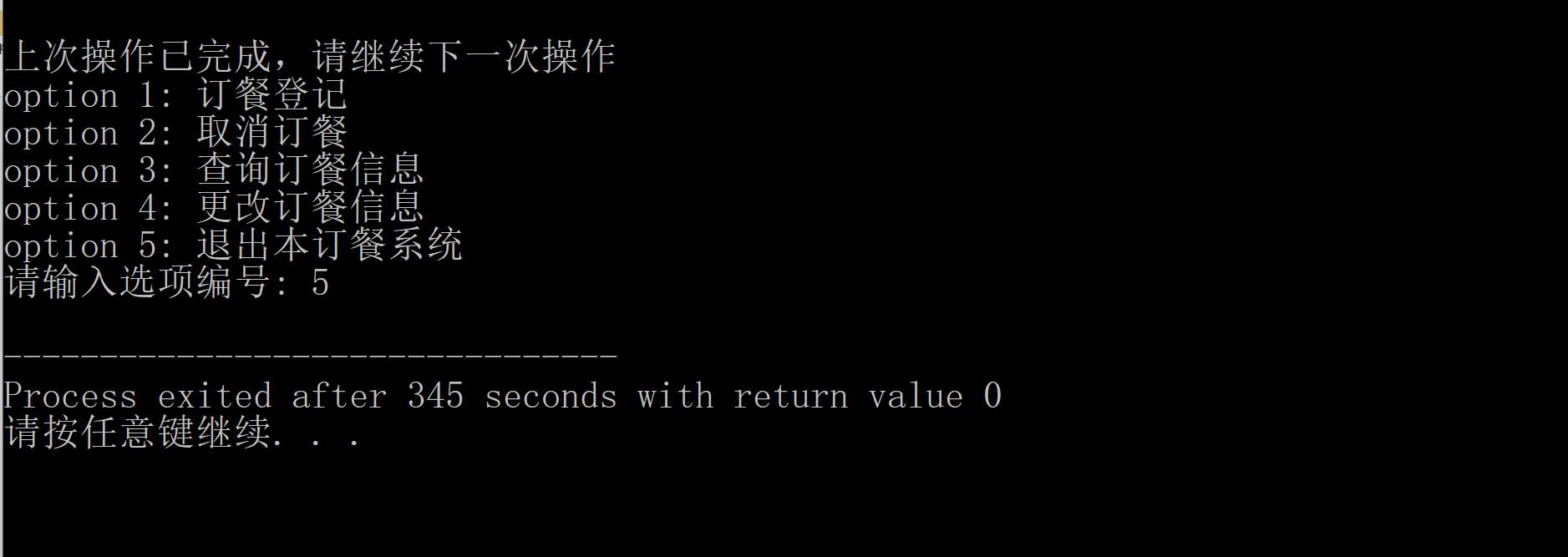
在TXT文件中表示如下



可以看到，中南小志君的订餐时间从2019/12/32/10/00更改为2025//11/1/12/00，用餐人数从20更改为15

## 功能五：退出订餐系统

用户在输入选项5后，退出订餐系统，此时按任何键终止此程序



# 源代码

## //main.c

#include "meal\_tele\_fun.h"

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char \*argv[]) {

Info guest[101];//餐厅最多接受100份订餐，1-100

int guest\_number=0;//编号，每次订餐登记时加一 ,从文件读入

guest\_number=read(guest);

interface();//绘出界面

int option=0;

while(1)//选择，在用户输入非法选项时提醒重新输入，一次操作完成后给出下一步操作提示

{

scanf("%d",&option);

do

{

switch(option)

{

case 1: {guest\_number++; meal\_regi(guest,guest\_number);};break;

case 2: meal\_canc(guest);break;

case 3: meal\_inqu(guest);break;

case 4: meal\_chan(guest);break;

case 5: return 0;

default: printf("请重新输入正确的编号: ");break;

}

}while(option!=1 && option!=2 && option!=3 && option!=4 && option!=5);

// system("CLS");

printf("\n\n上次操作已完成，请继续下一次操作\n");

printf("option 1: 订餐登记\n");

printf("option 2: 取消订餐\n");

printf("option 3: 查询订餐信息\n");

printf("option 4: 更改订餐信息\n");

printf("option 5: 退出本订餐系统\n");

printf("请输入选项编号: ");

}

return 0;

}

## //struct\_info

typedef struct mystruct

{

int number;//编号

char name[30];

long tele;

int peo\_num;//用餐人数<=20

char time[30];

char like[100];//用餐喜好，动态分配内存

}Info;

## //meal\_tele\_func.h

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include "struct\_info.h"

void meal\_rewr(Info guest[]);

int read(Info guest[]);

void dele\_gue(Info guest[],int guest\_number);

void meal\_canc\_option(Info guest[],int i);

void print(Info guest);

int compare(Info guest[]);

void interface();

void meal\_regi(Info guest[],int guest\_number);

void meal\_canc(Info guest[]);

void meal\_inqu(Info guest[]);

void meal\_chan(Info guest[]);

## //meal\_tele\_fun.c

#include "meal\_tele\_fun.h"

int read(Info guest[])//从txt文件中将顾客数据读入到结构体，返回最后一位顾客的编号

{

int guest\_number=0;

FILE \*fp=fopen("C:\\Users\\anhuishuangqiao\\Desktop\\C语言课设\\guest\_info.txt","a+");

if(NULL==fp)

{

printf("fopen fail in read");

return 0;

}

for(guest\_number=1;guest\_number<=100;guest\_number++)

{

fscanf(fp,"%s",guest[guest\_number].name);

fscanf(fp,"%ld",&guest[guest\_number].tele);

fscanf(fp,"%d",&guest[guest\_number].peo\_num);

fscanf(fp,"%s",guest[guest\_number].time);

fscanf(fp,"%s",guest[guest\_number].like);

fscanf(fp,"%d",&guest[guest\_number].number);

if(guest[guest\_number].tele==0 && strcmp(guest[guest\_number].name,"0")==0)//遇到被删除的内容，即0 0 1，指针退格再读

{

dele\_gue(guest,guest\_number);//利用dele\_gue的删除、重新写入的功能，逐个排除已被删除的内容

read(guest);//重新从文件读入结构体

// fseek(fp,strlen(time)+strlen(like)+5,SEEK\_CUR);

// continue;

}

if(guest[guest\_number].tele==0) break;//由于tele总是被初始化为0，故碰到0时，即停止读入剩下的空结构体

// print(guest[guest\_number]);

// printf("\n");

}

// print(guest[1]);

fclose(fp);

return \*&guest\_number-1;//由于其实多读了一个，所以减一

}

void dele\_gue(Info guest[],int guest\_number)//先删除结构体数组中的指定内容，再将结构体重新写入

{

for(int j=0;j<30;j++)//先清除对应结构体

{

guest[guest\_number].name[j]=0;

guest[guest\_number].time[j]=0;

guest[guest\_number].like[j]=0;

}

guest[guest\_number].tele=0;

guest[guest\_number].peo\_num=0;

FILE \*fp=fopen("C:\\Users\\anhuishuangqiao\\Desktop\\C语言课设\\guest\_info.txt","w+");

if(NULL==fp)

{

printf("fopen fail in dele\_gue");

return;

}

for(int k=1;k<=guest\_number-1;k++)//将除了删除项的项都存入新的txt文件

{

fprintf(fp,"%s %ld %d %s %s %d\n",guest[k].name, guest[k].tele, guest[k].peo\_num, guest[k].time, guest[k].like, guest[k].number);

}

for(int k=guest\_number+1;k<=100;k++)

{

if(guest[k].tele==0 && guest[k].peo\_num==0 && guest[k].number==0)//遇到尚未存入新信息的项就截止

{

break;

}

fprintf(fp,"%s %ld %d %s %s %d\n",guest[k].name, guest[k].tele, guest[k].peo\_num, guest[k].time, guest[k].like, guest[k].number);

}

fclose(fp);

}

void meal\_canc\_option(Info guest[],int i)//读入meal\_canc里的option选项，根据选项进行功能切换，i为符合输入的顾客编号

{

int option=0;

scanf("%d",&option);

if(1==option)

{

dele\_gue(guest,i);

}

else if(2==option)

{

meal\_canc(guest);

}

else if(3==option)

{

system("CLS");

interface();

}

else

{

printf("请重新输入正确的选项: ");

meal\_canc\_option(guest,i);

}

}

void print(Info guest)

{

printf("姓名: %s\n",guest.name);

printf("电话号码: %ld\n",guest.tele);

printf("用餐人数: %d\n",guest.peo\_num);

printf("用餐时间: %s\n",guest.time);

printf("用餐喜好: %s\n",guest.like);

}

int compare(Info guest[])//根据输入的姓名和电话号码寻找符合的顾客信息，输出信息，返回顾客编号

{

printf("\n请输入您的姓名: ");

char name[30];

scanf("%s",name);

printf("请输入您的电话号码: ");//防止同名

long tele;

scanf("%ld",&tele);

int i=0;

for(i=1;i<=100;i++)

{

if(strcmp(name,guest[i].name)==0 && tele==guest[i].tele)

{

printf("\n请核对您的信息");

print(guest[i]);

return \*(&i);

}

}

printf("对不起，系统未检测到您的信息");

return 0;

}

void interface()//初始界面

{

printf("-------------------------欢迎使用升华酒店电话订餐系统----------------------------\n");

printf("请按编号输入您的要求，我们将竭诚为您服务\n");

printf("option 1: 订餐登记\n");

printf("option 2: 取消订餐\n");

printf("option 3: 查询订餐信息\n");

printf("option 4: 更改订餐信息\n");

printf("option 5: 退出本订餐系统");

printf("\n请输入选项编号: ");

}

void meal\_regi(Info guest[],int guest\_number)//订餐登记，将信息存入结构体，再存到txt文件中去，注意地址

{

guest[guest\_number].number=guest\_number;

printf("请输入您的姓名： ");

scanf("%s",guest[guest\_number].name);

printf("请输入您的电话号码: ");

scanf("%ld",&guest[guest\_number].tele);

printf("请输入用餐人数: ");

scanf("%d",&guest[guest\_number].peo\_num);

printf("请输入用餐时间:(y/m/d/h/min) ");

scanf("%s",guest[guest\_number].time);

printf("请输入您的用餐喜好: ");

scanf("%s",guest[guest\_number].like);

FILE \*fp=fopen("C:\\Users\\anhuishuangqiao\\Desktop\\C语言课设\\guest\_info.txt","a");

if(NULL==fp)

{

printf("fopen fail in meal\_regi");

return;

}

//向文件依次写入 姓名 电话号码 用餐人数 用餐时间 用餐喜好 编号（放最后，方便读取）

fprintf(fp,"%s %ld %d %s %s %d\n",guest[guest\_number].name, guest[guest\_number].tele, guest[guest\_number].peo\_num, guest[guest\_number].time, guest[guest\_number].like, guest\_number);

fclose(fp);

}

void meal\_canc(Info guest[])//取消订餐

{

int i=compare(guest);//符合输入的顾客编号

if(0==i)

{

return;

}

printf("请问是否确认取消此次订餐(确认请输入1，信息错误想重新填写请输入2，否并返回主菜单请输入3\n");

printf("请输入选项: ");

meal\_canc\_option(guest,i);

}

void meal\_inqu(Info guest[])

{

printf("\n选择姓名请输入1，选择电话号码请输入2\n");

printf("请输入选项: ");

int option=0;

scanf("%d",&option);

char name[30];

long tele;

if(1==option)

{

printf("请输入姓名: ");

scanf("%s",name);

for(int i=1;i<=100;i++)

{

if(strcmp(name,guest[i].name)==0)

{

print(guest[i]);

return;

}

}

printf("对不起，系统未检测到您的信息");

}

else if(2==option)

{

printf("请输入电话号码: ");

scanf("%ld",&tele);

for(int i=1;i<=100;i++)

{

if(tele==guest[i].tele)

{

print(guest[i]);

return;

}

}

printf("对不起，系统未检测到您的信息");

}

else

{

printf("请重新输入正确的选项!\n");

meal\_inqu(guest);

}

}

void meal\_rewr(Info guest[])

{

FILE \*fp=fopen("C:\\Users\\anhuishuangqiao\\Desktop\\C语言课设\\guest\_info.txt","w+");

if(NULL==fp)

{

printf("fopen fail in meal\_rewr");

return;

}

for(int k=1;k<=100;k++)

{

if(guest[k].tele==0 && guest[k].peo\_num==0 && guest[k].number==0)//遇到尚未存入信息的项就截止

{

break;

}

fprintf(fp,"%s %ld %d %s %s %d\n",guest[k].name, guest[k].tele, guest[k].peo\_num, guest[k].time, guest[k].like, guest[k].number);

}

fclose(fp);

}

void meal\_chan(Info guest[])

{

printf("更改就餐时间请输入1，更改就餐人数请输入2，都更改请输入3\n");

printf("请输入选项: ");

int option=0;

scanf("%d",&option);

int i=compare(guest);

if(0==i)return;//未查询到信息时退出

if(1==option)

{

printf("请输入修改后的时间: ");

char new\_time[30];

scanf("%s",new\_time);

strcpy(guest[i].time,new\_time);

meal\_rewr(guest);

}

else if(2==option)

{

printf("请输入修改后的就餐人数: ");

int new\_num=0;

scanf("%d",&new\_num);

guest[i].peo\_num=new\_num;

meal\_rewr(guest);

}

else if(3==option)

{

printf("请输入修改后的时间: ");

char new\_time[30];

scanf("%s",new\_time);

strcpy(guest[i].time,new\_time);

printf("请输入修改后的就餐人数: ");

int new\_num;

scanf("%d",&new\_num);

guest[i].peo\_num=new\_num;

meal\_rewr(guest);

}

else

{

printf("请重新输入正确的选项: ");

}

}