

南京信息工程大学食堂点餐预约系统

陈宇恒

南京信息工程大学计算机学院、网络空间安全学院，江苏 南京 210044

摘要：

大学之中，吃饭难，一直是困扰广大师生的问题。所存在问题有三。第一，排队长。第二，座位难求。第三，用餐时间，食堂工作繁忙。为了解决以上就餐痛点，在互联网技术的发展下，《南京信息工程大学食堂点餐预约系统》应运而生。本文便是基于c++设计了《南京信息工程大学食堂点餐预约系统》，并对其进行了分析与探讨。

关键字：

点餐；c++；面向对象设计；管理系统；座位预约

1 项目目标

南京信息工程大学食堂点餐预约系统，能够让师生提前预约就餐时间段，能够提前预约自己心仪位置，利用好食堂的座位资源。食堂店铺人员可以管理菜单。店铺人员接到预约后可以根据订单提前备餐，防止食堂工作时间资源挤兑。管理员可以管理座位地图。

2 需求分析

2.1 顾客用户

为了方便师生用餐。师生可以线上提前预约座位,用餐时间段,预约菜品。查看座位情况等。

师生完成订单支付后，后台可以查看订单目前的状态。

师生使用该系统，没有账号的需要使用手机号进行注册，每一个手机号目前只支持注册一个账号。

2.2 商家客户

商家同上需要进行账号注册，方可使用该系统。

商家的主要功能是对菜谱的管理，以及接到订单后对订单的处理。

每一笔订单做好后，都应在后台数据中进行订单状态的修改，以反馈给顾客客户。

2.3 管理员

管理员的主要功能是对地图的管理。

管理员的账号和密码是学校下发的。目前不支持注册功能。

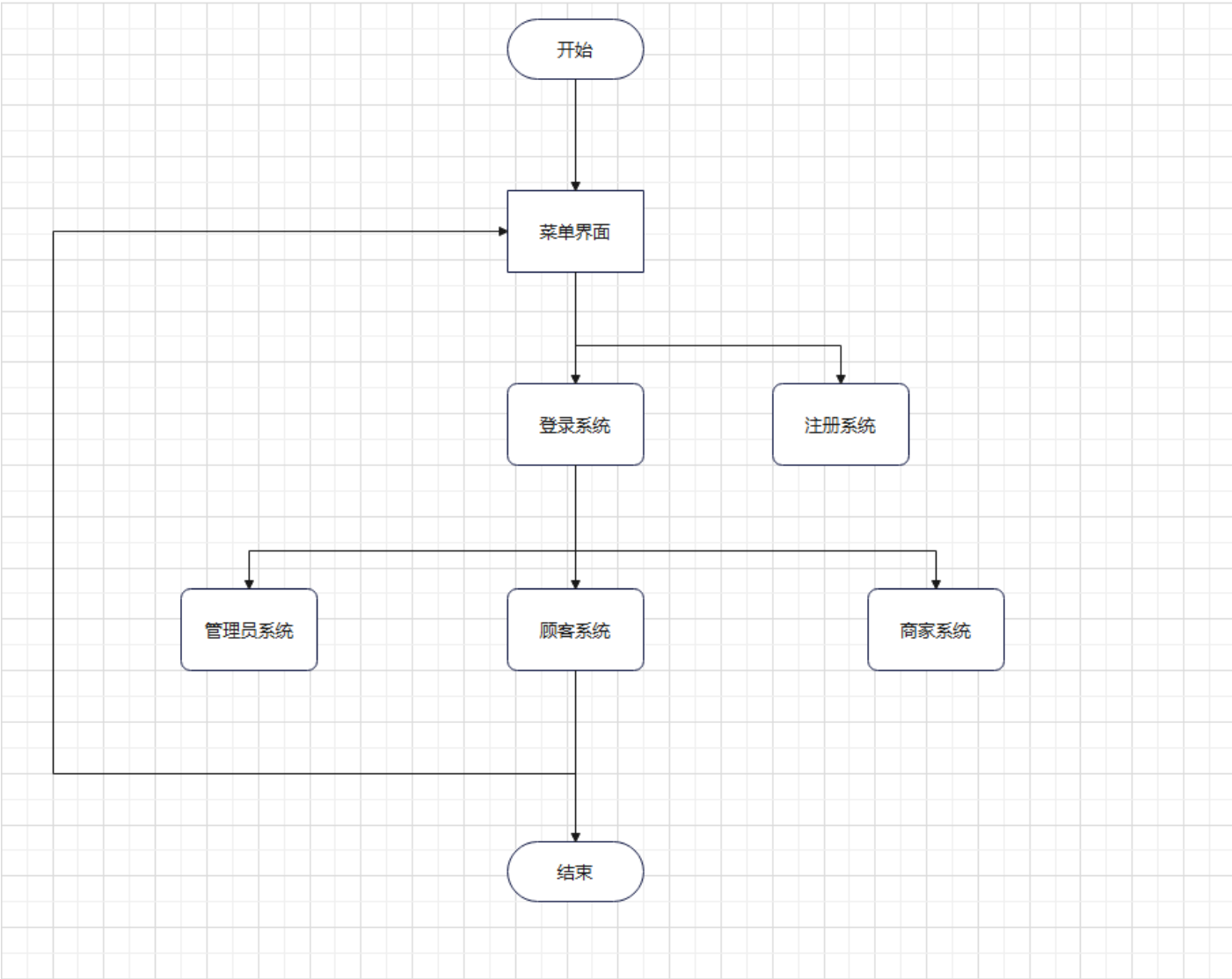
目前暂定的本系统为演示。

下发账号为:1

下发密码为:1

3 系统设计

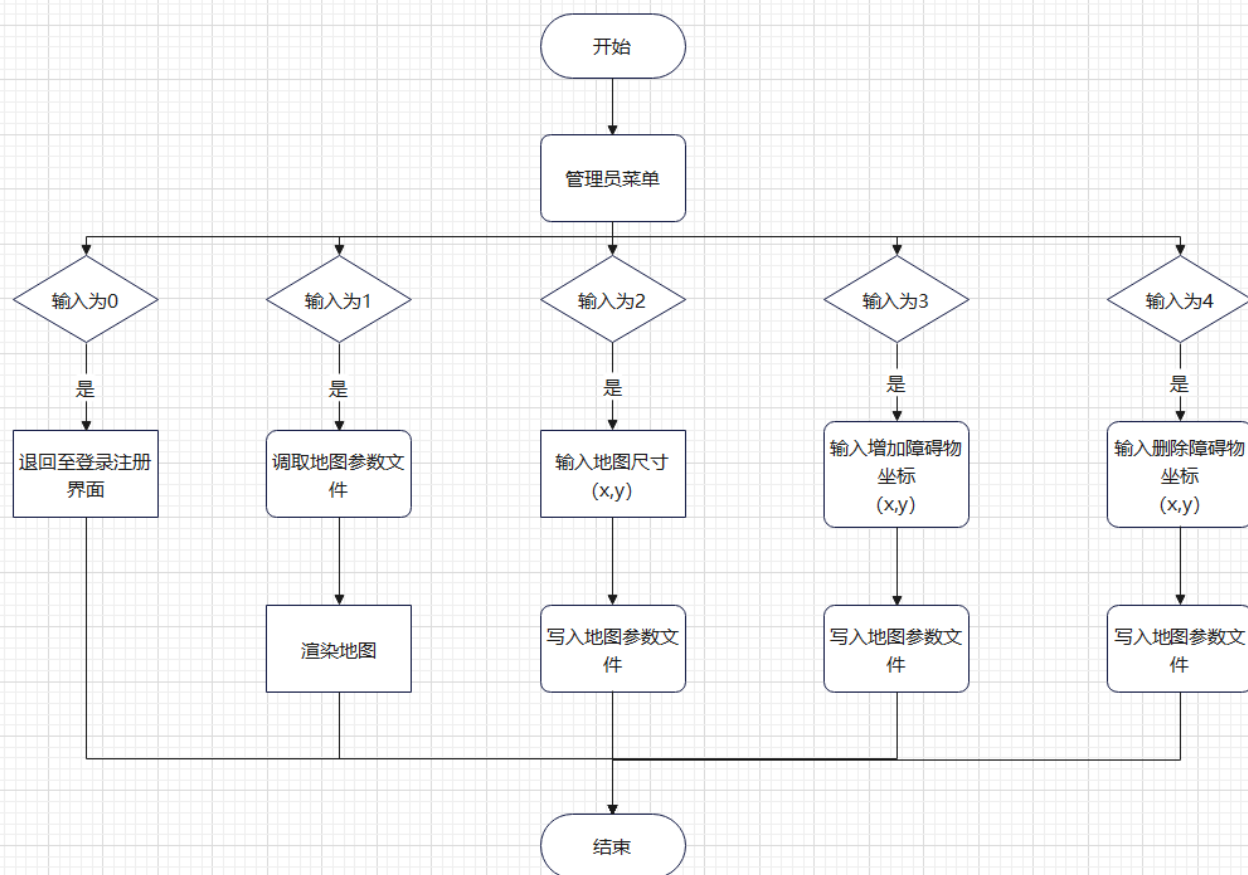
3.1 系统总体设计



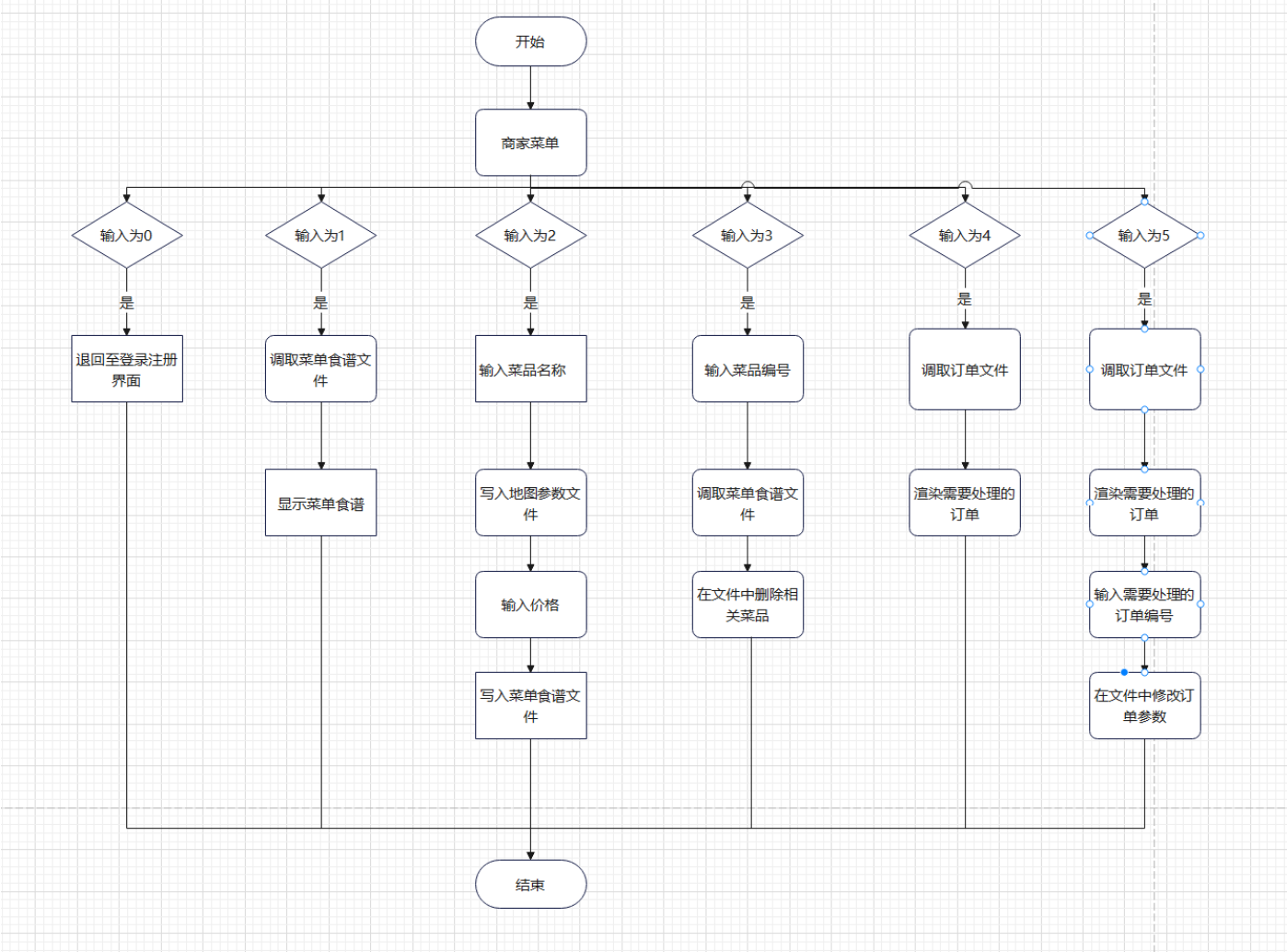
3.2 登录注册系统



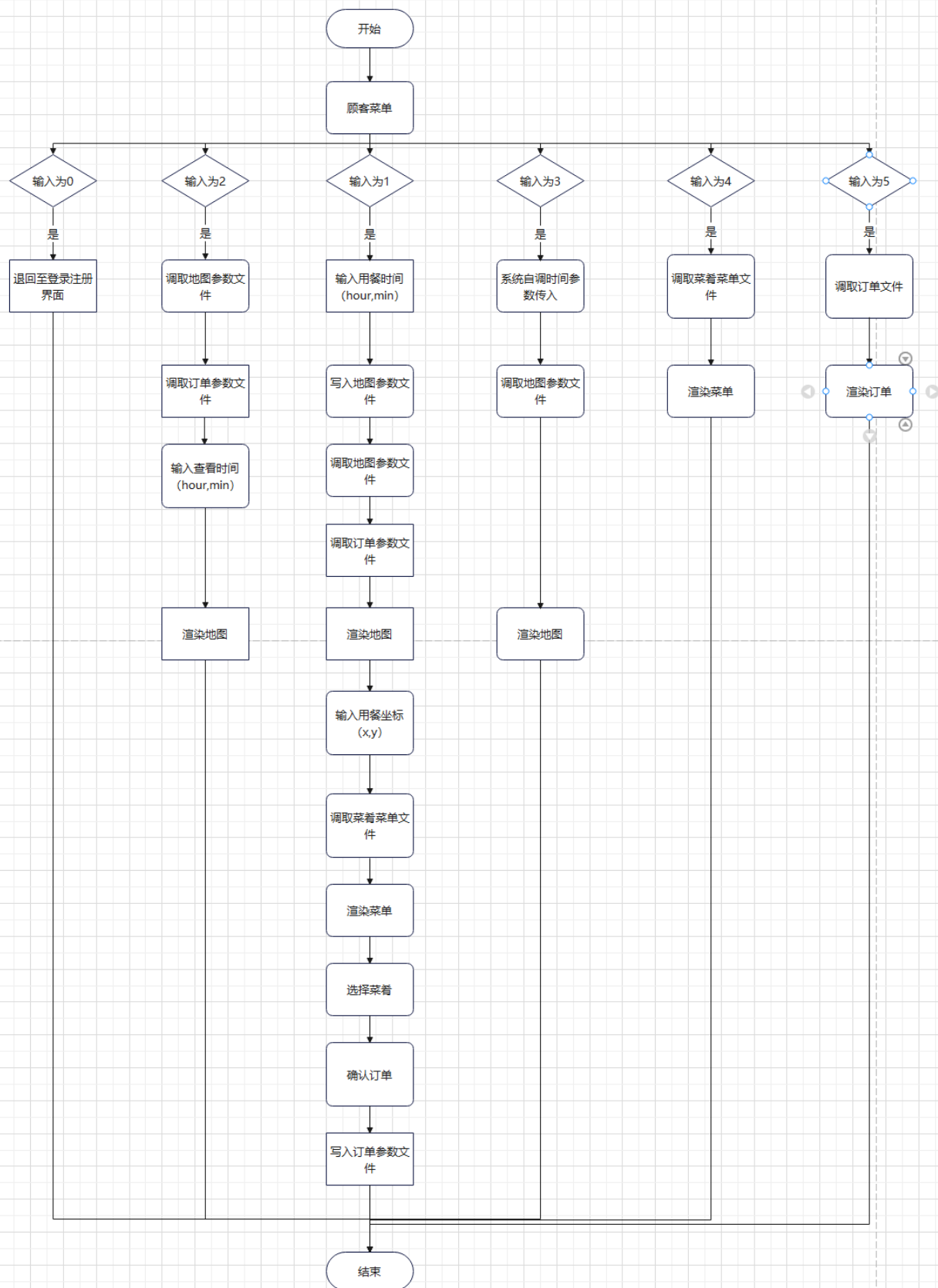
3.3 管理员系统



3.4 商家系统

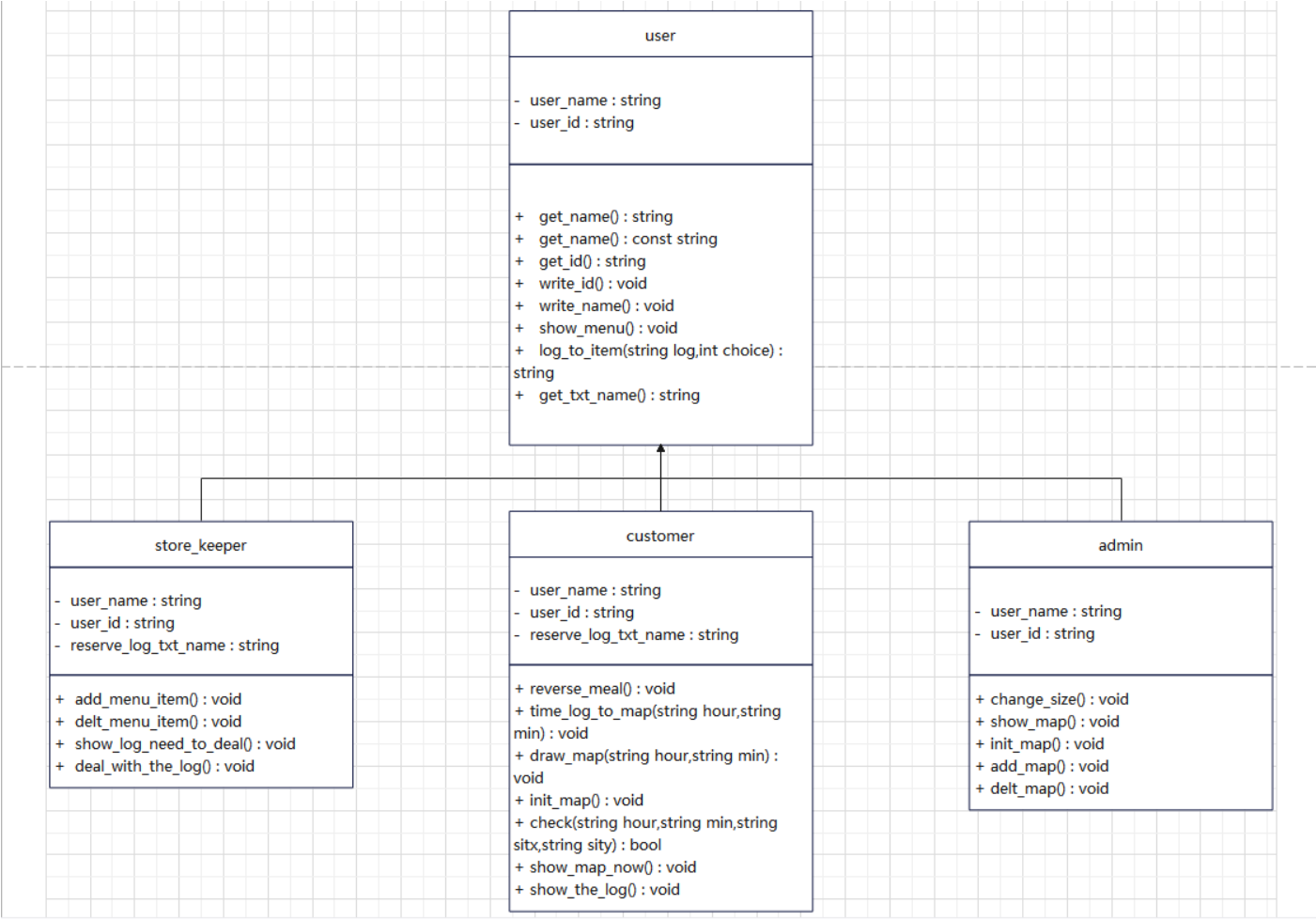


3.5顾客系统



4 类的设计

4.1 UML类图



4.2 类的说明和分析

4.2.1 user类

user 类是总的基类，具有一般的功能。

其中user_name和user_id是用户的用户名和账号。

void show_menu() 提供了从menu.txt文件里读出并打印订单的接口。

string log_to_item(string log,int choice)是将每一条订单记录的信息解析出来(详细介绍于核心设计)

string get_txt_name(): 每天自动调用系统日期，用于命名订单文件名。

4.2.2 store_keeper类

reserve_log_txt_name()调取写入订单文件的文件名。详细见核心设计。

void add_menu_item() 提供了增加菜品条目的接口。

void delt_menu_item() 提供了删除菜品条目的接口。

void show_log_need_to_be_deal() 展示没有被处理的订单。

void deal_with_the_log() 提供了处理订单的接口。

4.2.2 customer类

void reserve_meal() 核心函数,订单预定接口。

void time_log_to_map(string hour,string min)地图渲染引擎的组件。主要功能筛选在(hour,min) 的时间段内 的订单。并将他们的坐标取出来放入vector中保存。该插件实现了从三维地图到二维地图的降维。

void init_map()地图渲染引擎的组件。实现了管理员设置的地图信息的加载。

void draw_map(string hour,string min)地图渲染引擎主件画出地图。

void check(string hour,string min,string sitx,string sity) 实时的写权限检查，用于订单确认前最后一次检查权限。

void show_the_log()索引当前用户的订单。

4.2.3 admin类

void change_size() 改变地图大小参数

void add_map() 增加障碍物

void delt_map() 删除障碍物

void show_map() 展示管理员视角的地图

5 核心设计

5.3 系统自调时间和日期

使用"ctime"头文件调用time类。设计string get_time () 函数;自动调用系统时间，并且将时间转化为string类。

该核心功能用于订单文件系统的构建以及界面设计和美化。

```
string get_time()
{
    time_t now = time(NULL);
    tm* tm_t = localtime(&now);
```

```

std::stringstream ss;
ss << tm_t->tm_year + 1900 << "-";
if(tm_t->tm_mon + 1 < 10) ss << 0;
ss << tm_t->tm_mon + 1 << "-";
if(tm_t->tm_mday < 10) ss << 0;
ss << tm_t->tm_mday << " ";
if(tm_t->tm_hour < 10) ss << 0;
ss << tm_t->tm_hour << ":";
if(tm_t->tm_min < 10) ss << 0;
ss << tm_t->tm_min << ":";
if(tm_t->tm_sec < 10) ss << 0;
ss << tm_t->tm_sec;

return ss.str();
}

```

5.2 文件系统设计与信息交互

5.2.1 总体文件系统框架

名称		修改日期	类型	大小
	reserve_log	2022/12/22 17:35	文件夹	
	Account_system_admin.txt	2022/12/4 22:52	文本文档	1 KB
	Account_system_customer.txt	2022/12/20 15:29	文本文档	1 KB
	Account_system_storekeeper.txt	2022/12/16 13:43	文本文档	1 KB
	canteen_meal_reverse_system.exe	2022/12/20 15:25	应用程序	2,107 KB
	index.txt	2022/12/20 15:33	文本文档	1 KB
	map.txt	2022/12/20 15:28	文本文档	1 KB
	menu.txt	2022/12/20 15:33	文本文档	1 KB

5.2.2 账号系统文件

账号系统主要是

Account_system_admin.txt

Account_system_store_keeper.txt

Account_system_customer.txt

三个文件中。分别对应管理员，商家，顾客。

文件里使用空格符隔开，分别是账号，密码，用户名，注册的手机号

```
Account_system_customer.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
20220013 12345 cyhcyhcyh 13405862003
20220012 12345 abc 13405862006
20220011 12345 ccdd 13405862003
20220010 123 1123 11331
20220009 12345 ccc 13405862004
20220008 12345 cyhcyh 13405862003
20220007 200366 huihui 13851521427
20220005 12345 cctalk 15150678650
20220004 12345 cyhh 134434
2 2 cyh 13405862003
end
```

5.2.3 地图参数文件

地图参数文件即 map.txt;

每一行两个数字。空格隔开。第一行是地图的大小参数。

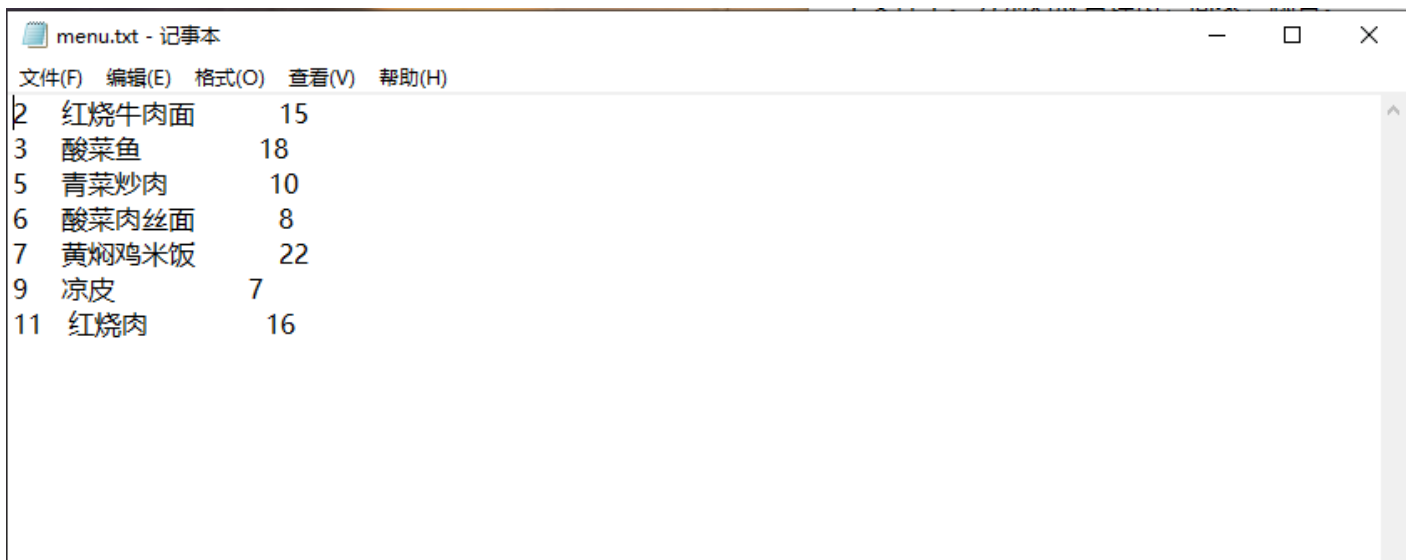
其余是地图障碍物横纵坐标参数。

```
map.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
10 13
3 3
3 4
3 5
4 3
4 4
4 5
3 8
4 8
7 8
7 9
7 10
8 8
8 9
8 10
```

5.2.4 菜单文件

菜单文件即 menu.txt;

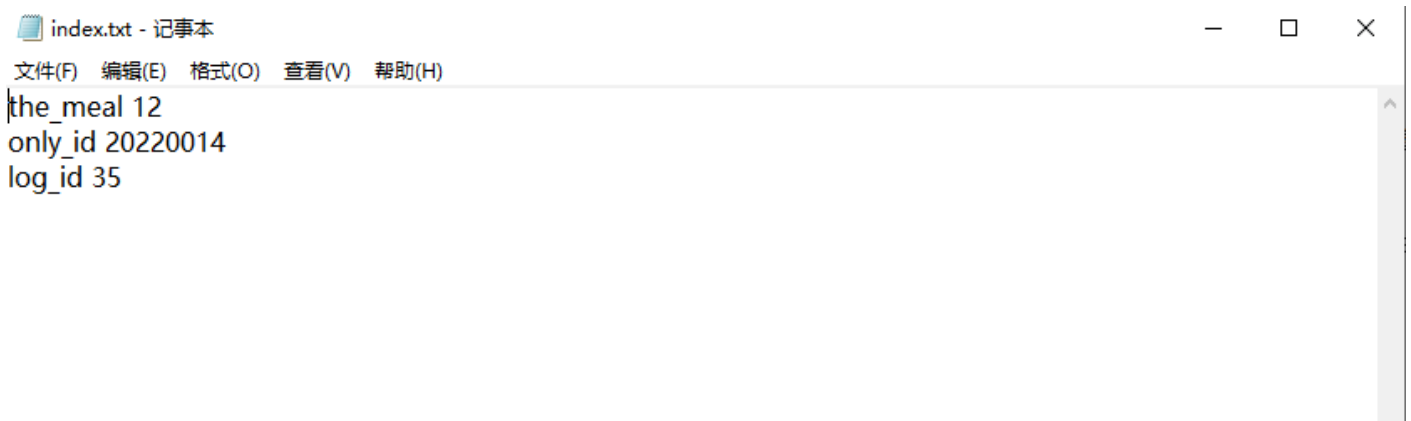
每一列分别是菜品id号,菜品名称,价格



5.2.5 自动分配id

每次向文件写入条目的时候都会自动调用

index.txt文件



index.txt文件中存储了相应的序号

the_meal 对应的是餐品id

only_id 对应的是自动分配的用户账号id

log_id 对应的是为每一笔订单分配的唯一索引id

自动分配id的核心代码如下

```
ifstream write_id;
    ofstream write;
    string ids;int id;
    queue<string> temp;
    write_id.open("index.txt");

    while(getline(write_id,line))
    {
        int pos1 = line.find(" ") + 1;
        int pos2 = line.find(" ",pos1);
        string lines = line.substr(0, pos1 - 1);
```

```

    if(lines != "log_id") temp.push(line);
    else
    {
        ids = line.substr(pos1, pos2 - pos1) ;
        istringstream ss(ids);
        ss >> id;
        line = lines + " " + to_string(id + 1);
        temp.push(line);
    }

}

write_id.close();

//id
write.open( "index.txt",ios::in | ios::trunc);

while(!temp.empty())
{
    if(temp.front() != " ")
        write << temp.front() << endl;
    temp.pop();
}

write.close();

```

程序内部拿到的是string类的ids，为当前分配的id。

5.2.6 订单文件系统

在reserve_log文件夹下,系统每次调用xxxx-xx-xx_reserve_log.txt这一类文件时,会自己调用日期。

第一，如果reserve_log文件夹没有相应日期的txt文件，系统会自己创建xxxx-xx-xx_reserve_log.txt文件。

第二,如果存在相应文件，便在后面追加写。

该核心功能解决了三个问题。

第一，每天打开一个新的文件，防止在地图渲染引擎造成数据冗余(指的是不同日期相同时间段的数据会给引擎造成误判)。每天新开文件有利于本软件在时间的使用上具有可持续性。

第二,自服务开始的所有订单数据都有服务器备份。这就表明，在订单出现争议的时候,做到有据可查。

第三,后期可以提供大数据支持，计算菜品的经济效益。

食堂点餐系统... > reserve_log		在 reserve_log 中搜索		
名称	修改日期	类型	大小	
2022-12-17_reserve_log.txt	2022/12/17 18:35	文本文档	1 KB	
2022-12-19_reserve_log.txt	2022/12/19 18:11	文本文档	1 KB	
2022-12-20_reserve_log.txt	2022/12/20 15:34	文本文档	1 KB	

订单文件与主程序的信息交互

14 50 7 1 6	酸菜肉丝面	8	cyh 2 20 0
14 56 5 3 2	红烧牛肉面	15	cyh 2 21 0
14 56 4 3 2	红烧牛肉面	15	cyh 2 22 01
16 30 3 6 3	酸菜鱼	18	ccc 20220009 23 0
16 30 4 6 3	酸菜鱼	18	ccc 20220009 24 0
16 30 4 2 6	酸菜肉丝面	8	cyh 2 25 01
16 30 3 2 2	红烧牛肉面	15	cyh 2 26 0
12 00 3 2 2	红烧牛肉面	15	cyh 2 27 0
1 1 1 1 2	红烧牛肉面	15	cyh 2 28 0
18 30 8 5 4	土豆烧鸡	14	abc 20220012 29 0
18 30 6 8 7	黄焖鸡米饭	22	abc 20220012 30 0
18 45 5 4 6	酸菜肉丝面	8	cyh 2 31 01
19 00 7 11 7	黄焖鸡米饭	22	cyh 2 32 01
20 00 5 11 9	凉皮	7	cyh 2 33 0

以上为后台订单信息。

每一笔订单都以行的形式写在当日的reserve_log.txt文件当中。

每一行的订单按序包括

用餐小时, 用餐分钟, 用餐地址 (横坐标, 纵坐标), 菜肴名称, 订单金额, 用户名, 账号, 订单id, 订单状态

每一笔订单以固定的格式写入文件中。因此也实现了解析每一笔订单以上各个条目的接口。

每一行订单的每个条目解析出来都是一个string串。

后续程序可以根据需要调取不同的条目。

以下为核心解析代码。函数为基类user类成员。

```
string user::log_to_item(string log,int choice)
{
    int pos1 = log.find(" ") + 1; string show_time_hour = log.substr(0, pos1 - 1);

    int pos2 = log.find(" ",pos1) + 1; string show_time_min = log.substr(pos1,pos2 - pos1);
    int pos3 = log.find(" ",pos2) + 1; string show_x = log.substr(pos2,pos3 - pos2);
    int pos4 = log.find(" ",pos3) + 1; string show_y = log.substr(pos3,pos4 - pos3);
    int pos5 = log.find(" ",pos4) + 4;
    int pos6 = log.find(" ",pos5) + 1; string meal_name = log.substr(pos5,pos6 - pos5);
    int pos7 = pos5 + 20;
```

```

int pos8 = log.find(" ",pos7) + 1; string show_money = log.substr(pos7,pos8 - pos7);
int pos9 = pos7 + 9;
int pos10 = log.find(" ",pos9) + 1; string show_name = log.substr(pos9,pos10 - pos9);
int pos11 = log.find(" ",pos10 + 1) + 1; string show_id = log.substr(pos10,pos11 - pos10);
int pos12 = log.find(" ",pos11) + 1; string show_log_id = log.substr(pos11,pos12 - pos11);
int pos13 = log.find(" ",pos12) + 1; string show_status = log.substr(pos12,pos13 - pos12);

if(choice == 1)          return show_time_hour;
else if(choice == 2)      return show_time_min;
else if(choice == 3)      return show_x;
else if(choice == 4)      return show_y;
else if(choice == 5)      return meal_name;
else if(choice == 6)      return show_money;
else if(choice == 7)      return show_name;
else if(choice == 8)      return show_id;
else if(choice == 9)      return show_log_id;
else if(choice == 10)     return show_status;
}

```

5.3 简单地图渲染引擎设计

5.3.1 初始化地图

```

const int N = 1000;
memset (is_set,false,sizeof is_set);
for(int i = 0;i < N;i ++ )
for(int j = 0;j < N;j ++ )
    maps[i][j] = '0';

```

5.3.2 调取地图参数文件

用于加载管理员版本的地图信息。

map_x, map_y: 记录了加载到程序中的地图宽和长。

```

bool f = true;

while(getline(read,map))
{
    int pos1 = map.find(" ") + 1;
    int pos2 = map.find(" ",pos1);

    string sxx = map.substr(0, pos1 - 1);
    string syy = map.substr(pos1, pos2 - pos1) ;
    istringstream xxx(sxx);
    istringstream yyy(syy);
    int x_int; xxx >> x_int;
    int y_int; yyy >> y_int;
    if(f)
    {
        mapx = x_int;
        mapy = y_int;
        f = false;
    }
}

```

```

    }
    else {
        maps[x_int][y_int] = '#';
        is_set[x_int][y_int] = true;
    }
}

read.close();

```

5.3.3 加载订单文件信息

本软件的地图是二维平面加上一维时间。实际上本软件的地图是三维的。

每一笔订单记录了时间地点，订单在地图层面也是三维的。

我们对于每一笔订单，以时间为维度，选择出在一个时间段的订单来。

并且根据索引加载他们的位置坐标。

将该订单的坐标加载出来，添加至地图中。

值得一提的是打印出来的地图是二维的，因为时刻被唯一确定了。

```

int h;int m;
int start_time;int end_time;
istringstream sh(hour);
istringstream sm(min);
sh >> h;
sm >> m;
start_time = h * 60 + m;
end_time = start_time + 20;

ifstream read;
read.open(reserve_log_txt_name);

string line;

while(getline(read,line))
{
    int pos1 = line.find(" ") + 1;
    int pos2 = line.find(" ",pos1);

    string new_hour = line.substr(0, pos1 - 1);
    string new_min = line.substr(pos1, pos2 - pos1) ;
    istringstream snh(new_hour);
    istringstream snm(new_min);
    int new_h; snh >> new_h;
    int new_m; snm >> new_m;
    int new_time_s = 60 * new_h + new_m;
    int new_time_e = 60 * new_h + new_m + 20;
}

```



```

        if( (new_time_s >= start_time && new_time_s <= end_time) ||
            (new_time_e >= start_time && new_time_e <= end_time) )
            now_log.push_back(line);

    }
    read.close();

    while(!now_log.empty())
    {

        string t = now_log.back();
        now_log.pop_back();

        string str_x = log_to_item(t,3);
        string str_y = log_to_item(t,4);

        istringstream xxx(str_x);
        istringstream yyy(str_y);
        int xx; xxx >> xx;
        int yy; yyy >> yy;

        now_map.push_back({xx,yy});
    }

```

5.3.3 渲染地图

所有信息加载完毕，便可以渲染地图了。

```

for(int i = 0;i <= mapx;i ++)
{for(int j = 0;j <= mapy;j ++)
{
    if(j == 0) cout << left << setw(3) << i ;
    else if(i == 0) cout << left << setw(3) << j;
    else cout << left << setw(3) << maps[i][j] ;
}
cout << endl;
}

```

5.4 界面和程序进程的优化设计

5.4.1 界面函数化调用

在主程序里面，每一个界面都被函数化，清晰而又优雅。而且方便组合使用

。再者主界面调用了时间函数，用户在使用的时候可以查看当前时间。

并且配备清屏功能，优雅。

例如

```

void head_show()
{
    system("cls");
    cout << "#####" << endl;
    cout << "南京信息工程大学 自助点餐预定系统欢迎您!      #" << endl;
    cout << "                                #" << endl;
    cout << "Welcome to canteen_meal_reserve_system of NUIST! #" << endl;
    cout << "#####" << endl;
    cout << "现在的时间是:                                #" << endl;
    cout << get_time() << "                                #" << endl;
    cout << "#####" << endl;
    return;
}

```

5.4.2 进程的优化设计

每一个进程包含当前进程界面的显示函数。和选择分支。

选择分支里包括下一个界面的入口和下一个界面的显示函数。

四位一体,拆分进程,是程序打开一次就可以拥有很好的持续使用性。

仅举如下例子加以说明

```

bool menu(int choice)
{
    if(!choice) return false;
    head_show();

    switch(choice){
        case 1:
            log_in();
            menu_show();
            cin >> choice;
            menu(choice);
            break;
        case 2:
            register_system();
            menu_show();
            cin >> choice;
            menu(choice);
            break;
        default:
            cout << "输入错误!" << endl;
            system("pause");
            menu_show();
            cin >> choice;
            menu(choice);
            break;
    }
}

```

```

void menu_show()
{
    head_show();
}

```

```
cout << "输入数字, 进行操作:"<< endl;  
cout << "0:退出软件" << endl;  
cout << "1:登录" << endl;  
cout << "2:注册" << endl;  
  
}
```

6 开发及调试

6.1 总体开发框架

系统的总基类user类。根据需要分为了admin, customer, store_keeper三个类别。

admin类核心功能是管理地图。如更改地图大小。增加障碍物（桌椅等）障碍物是不可以坐人的坐标。

customer类核心功能是预定订单,查看订单状态。

预定订单也是本软件的核心功能。该功能较为复杂。因此拆分为了几个小功能。

第一，查看座位情况，也就是渲染地图。该功能的开发实现详见第五章的简单地图渲染引擎设计。

第二, 继承自基类的查看菜谱功能。

第三，写订单前的容错检查。

store_keeper类实现的核心功能是对菜谱的增删改查，和对于订单的处理。

6.2 详细设计

6.2.1 主界面

```
选择 C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您！#
#
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在的时间是：#
2022-12-22 23:01:23#
#####
输入数字, 进行操作：#
0:退出软件#
1:登录#
2:注册#
_
```

6.2.2注册系统界面

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您！#
#
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在的时间是：#
2022-12-22 23:21:35#
#####
这里是注册系统：#
#
#####
输入数字，选择注册角色#
1: 顾客#
2: 商家#
_
```

6.2.3注册系统

注册系统核心代码简略如下

```
if(phone_number == in_number)
{

    cout << "当前号码已经被注册！" << endl;

    system("pause");
    return;

}

string names;
cout << "请输入您的用户名: "; cin >> names;

cout << "请设置密码: "; string set_passwords = set_pd();
cout << endl;
cout << "再次确认密码: "; string insure_passwords = set_pd();
if(set_passwords != insure_passwords)
```

```

{
    head_show();
    cout << "这里是注册系统:                                #" << endl;
    cout << "                                                #" << endl;
    cout << "#####" << endl;
    register_table(choice);
    cout << "两次密码不匹配, 请重新注册!" << endl;
    system("pause");
    return;
}

```

注册系统的密码不会回显。核心代码如下

```

string set_pd()
{
    string passwords;
    char pd;
    int i = 0;
    while(1)
    {
        pd = _getch();          //获取输入的密码
        if (pd == 13)          //回车返回
        {
            break;
        }
        if (pd == 8)           //退格
        {
            if (passwords.length() == 0)          //如果第一个键为退格键
            {
                cout << "密码为空, 请输入密码: " << endl;
                continue;
            }
            cout << "\b \b";
            i++;
            passwords[passwords.length() - i] = '\0';

        }
        else
        {
            passwords += pd;
            cout << "*";
        }
    }
    return passwords;
}

```

输入手机号码进行注册,如果当前手机号被注册过会有提示

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您! #  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是:  
2022-12-22 23:25:56 #  
#####  
这里是注册系统: #  
#  
#####  
现在正在进行顾客注册:  
#####  
请输入绑定的电话号码:13405862003  
当前号码已经被注册!  
请按任意键继续. . .
```

正常没有注册过的手机号注册如下

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您! #  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是:  
2022-12-22 23:30:01 #  
#####  
这里是注册系统: #  
#  
#####  
现在正在进行顾客注册:  
#####  
请输入绑定的电话号码:13405862005  
请输入您的用户名: cyyh  
请设置密码: *****  
再次确认密码: *****
```

两次密码不匹配会提示如下

选择 C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reserve_system.exe

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您！#  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是：#  
2022-12-22 23:34:06#  
#####  
这里是注册系统：#  
#  
#####  
现在正在进行顾客注册：#  
#####  
两次密码不匹配，请重新注册！  
请按任意键继续. . .
```

注册成功会提示如下。并且分配账号

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reserve_system.exe

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您！#  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是：#  
2022-12-22 23:36:28#  
#####  
这里是注册系统：#  
#  
#####  
现在正在进行顾客注册：#  
#####  
请输入绑定的电话号码:13405862005  
请输入您的用户名: cyyh  
请设置密码: ****  
再次确认密码: ****  
您的平台唯一账号是: 20220014  
请牢记！  
请按任意键继续. . .
```

6.2.4 登录系统

登录系统的菜单界面

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您！      #
                                                    #
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在是时间：                                      #
2022-12-22 23:48:02                                #
#####
输入数字，选择登录角色
1:管理员
2:顾客
3. 商家
```

登录系统的主界面 一共有三次登录机会

```
选择 C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您！      #
                                                    #
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在是时间：                                      #
2022-12-22 23:49:28                                #
#####
现在正在进行顾客登录：
#####
请输入您的账号：
20220014
请输入您的密码：
1234
```

登录系统核心代码

先找账号，若没有找到。

```
if (system_id == "end" && flagss)
{
    head_show();
    cnt ++;
    cout << "账号错误，没有找到该账号!" << endl;
    system("pause");
}
```

账号不正确会提示如下信息


```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您! #
#
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在的时间是: #
2022-12-23 10:34:53 #
#####
账号错误, 没有找到该账号!
请按任意键继续. . . █
```

账号匹配后, 再找匹配密码。

密码匹配失败如下

cnt用来记录尝试次数, 如若超过三次, 会跳出界面。

```
else
{
    head_show();
    flagss = false;
    cout << "密码错误!" << endl;
    cnt ++;
    system("pause");
}
```

密码错误会提示如下信息

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您! #
#
Welcome to canteen_meal_reserve_system of NUIST! #
#####
现在的时间是: #
2022-12-22 23:52:08 #
#####
密码错误!
请按任意键继续. . .
```

登录次数也会做相应的调整

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您! #  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是: #  
2022-12-22 23:52:45 #  
#####  
现在正在进行顾客登录:  
#####  
请您再次尝试, 一共3次机会!  
目前尝试了2次  
请输入您的账号:  
12313123_
```

密码匹配成功如下。

```
if (system_id == id) //登录成功  
    if (system_password == password)  
    {  
        int pos1 = line.find(" ") + 1;        string only_id = line.substr(0, pos1 - 1);  
        int pos2 = line.find(" ", pos1) + 1; string passwords = line.substr(pos1, pos2 - pos1);  
        int pos3 = line.find(" ", pos2) + 1; string the_name = line.substr(pos2, pos3 - pos2);  
        if(choice == 3) store_keeper_system(only_id, the_name);  
        else if(choice == 2) customer_system(only_id, the_name);  
        else if(choice == 1) admin_system(only_id, the_name);  
    }  
    return;  
}
```

6.2.5 管理员系统

6.2.5.1 管理员系统界面

在界面上输入相应的数字, 可以进行相应的操作

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
#####
南京信息工程大学 自助点餐预定系统欢迎您! #
#
Welcome to canteen_meal_reserve_system of NUIST! #
#####
当前的时间是: #
2022-12-23 11:53:14 #
#####
这里是管理员客户端
#####
hhq管理员, 欢迎您!
#####
输入数字, 进行操作:
0:退出系统
1:查看地图
2:修改地图大小
3:增加障碍物
4:删除障碍物
_
```

6.2.5.2 修改地图尺寸

输入相应的参数,可以修改地图的宽和高。

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
请输入更改后地图的大小参数
x:13
y:14_
```

修改后可以通过查看地图功能查看效果。

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
*****
* '#' 代表目前的障碍物或者桌子等, 是不可以坐人的位置。 *
* '0' 代表目前的座位或者椅子, 是可以坐人的位置。 *
*****
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 # 0 0 0 0 0
4 0 0 # # # 0 0 # 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 0 0 0 0 0 0 0 0 0 0
7 0 0 0 0 0 0 0 0 # # # 0 0 0
8 0 0 0 0 0 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请按任意键继续. . .
```

核心思想是在map.txt文件上操作。

正如前文5.2 节介绍，本操作代码的核心是修改map.txt文件的第一行参数。

在软件内部，map.txt文件的第一行参数被定义为地图的宽和高。

```
void admin::change_size()
{
    cout << "请输入更改后地图的大小参数" << endl;
    queue<string> temp;
    string x;
    string y;
    cout << "x:";cin >> y;
    cout << "y:";cin >> x;
    bool f = true;
    ifstream read;string line;
    read.open("map.txt",ios::in);
    while(getline(read,line))
    {
        if(f) f = false;
        else temp.push(line);
    }
    read.close();

    ofstream write;
    write.open("map.txt",ios::out|ios::trunc);
    write << x << ' ' << y << endl;

    while(!temp.empty())
    {
        write << temp.front() << endl;
        temp.pop();
    }

    write.close();
}
```

6.2.5.3 增删障碍物

首先输入需要增加障碍物的个数。

然后通过输入相应的横纵坐标参数，可以对地图的障碍物进行增减。

选择 C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
*****
* '#'代表目前的障碍物或者桌子等,是不可以坐人的位置。 *
* '0'代表目前的座位或者椅子,是可以坐人的位置。 *
*****
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 # 0 0 0 0 0
4 0 0 # # # 0 0 # 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 0 0 0 0 0 0 0 0 0 0
7 0 0 0 0 0 0 0 # # # 0 0 0
8 0 0 0 0 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请输入要加入几个点:
4
1:
x:3
y:7
2:
x:4
y:7
3:
x:3
y:8
4:
x:4
y:8
请按任意键继续. . .
```

添加后可以查看效果

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
*****
* '# 代表目前的障碍物或者桌子等, 是不可以坐人的位置。 *
* '0' 代表目前的座位或者椅子, 是可以坐人的位置。 *
*****
0  1  2  3  4  5  6  7  8  9  10 11 12 13
1  0  0  0  0  0  0  0  0  0  0  0  0  0
2  0  0  0  0  0  0  0  0  0  0  0  0  0
3  0  0  #  #  #  0  0  #  0  0  0  0  0
4  0  0  #  #  #  0  0  #  0  0  0  0  0
5  0  0  0  0  0  0  0  0  0  0  0  0  0
6  0  0  0  0  0  0  0  0  0  0  0  0  0
7  0  0  #  #  0  0  0  #  #  #  0  0  0
8  0  0  #  #  0  0  0  #  #  #  0  0  0
9  0  0  0  0  0  0  0  0  0  0  0  0  0
10 0  0  0  0  0  0  0  0  0  0  0  0  0
11 0  0  0  0  0  0  0  0  0  0  0  0  0
12 0  0  0  0  0  0  0  0  0  0  0  0  0
13 0  0  0  0  0  0  0  0  0  0  0  0  0
14 0  0  0  0  0  0  0  0  0  0  0  0  0
请按任意键继续. . .
```

如下展示删除操作

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
*****
* '# 代表目前的障碍物或者桌子等, 是不可以坐人的位置。 *
* '0' 代表目前的座位或者椅子, 是可以坐人的位置。 *
*****
0  1  2  3  4  5  6  7  8  9  10 11 12 13
1  0  0  0  0  0  0  0  0  0  0  0  0  0
2  0  0  0  0  0  0  0  0  0  0  0  0  0
3  0  0  #  #  #  0  0  #  0  0  0  0  0
4  0  0  #  #  #  0  0  #  0  0  0  0  0
5  0  0  0  0  0  0  0  0  0  0  0  0  0
6  0  0  0  0  0  0  0  0  0  0  0  0  0
7  0  0  #  #  0  0  0  #  #  #  0  0  0
8  0  0  #  #  0  0  0  #  #  #  0  0  0
9  0  0  0  0  0  0  0  0  0  0  0  0  0
10 0  0  0  0  0  0  0  0  0  0  0  0  0
11 0  0  0  0  0  0  0  0  0  0  0  0  0
12 0  0  0  0  0  0  0  0  0  0  0  0  0
13 0  0  0  0  0  0  0  0  0  0  0  0  0
14 0  0  0  0  0  0  0  0  0  0  0  0  0
请输入需要删除标记的次数:
2
1:
x:8
y:3
2:
x:8
y:4
请按任意键继续. . .
```

删除后可以查看效果

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
*****
* '#'代表目前的障碍物或者桌子等,是不可以坐人的位置。 *
* '0'代表目前的座位或者椅子,是可以坐人的位置。 *
*****
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 0 0 0 0 0 0
4 0 0 # # # 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 0 0 0 0 0 0 0 0 0 0
7 0 0 # # 0 0 0 # # # 0 0 0
8 0 0 # # 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请按任意键继续. . .
```

核心思想也是对于map.txt文件进行操作。以下仅展示增加障碍物的代码。

```
void admin::add_map()
{
    show_map();
    int n;
    cout << "请输入要加入几个点: " << endl;
    cin >> n;

    ofstream write;
    write.open("map.txt",ios::out|ios::app);

    string x;
    string y;

    for(int i = 1;i <= n;i ++){
        cout << i << ":" << endl;
        cout << "x:";cin >> y;
        cout << "y:";cin >> x;
        string log = x + ' ' + y;
        write << log << endl;
    }

    write.close();
}
```

6.2.6 顾客系统

6.3.6.1 顾客客户端主界面

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您! #  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是: #  
2022-12-23 12:42:30 #  
#####  
这里是顾客客户端  
#####  
cyyh , 欢迎您!  
#####  
输入数字, 进行操作:  
0:退出系统  
1:订单预定  
2:查看座位情况  
3:查看现在座位情况  
4:查看菜单  
5:查看订单状态
```

6.3.6.2 订餐

订餐系统主要用到了调用菜谱,简单地图渲染引擎。

首先选择用餐的时间。

```
现在正在点餐时间预定(1/4)  
请您选择用餐开始时间:  
用餐限定20分钟:  
时: 18  
分: 00  
请按任意键继续. . .
```

调用地图渲染引擎。


```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
现在正在点餐座位预定(2/4)
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 0 0 0 0 0 0
4 0 0 # # # 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 0 0 0 0 0 0 0 0 0 0
7 0 0 # # 0 0 0 # # # 0 0 0
8 0 0 # # 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请输入坐标，选择座位：
x: 4
y: 6
请按任意键继续. . .
```

调用菜谱。

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
现在正在点餐菜品预定(3/4)
序号 菜肴名称 价格（元）
2 红烧牛肉面 15
3 酸菜鱼 18
5 青菜炒肉 10
6 酸菜肉丝面 8
7 黄焖鸡米饭 22
9 凉皮 7
11 红烧肉 16
请选择菜品编号：
3
```

确认订单。

```
现在正在点餐预定(4/4)
正在进行订单确认
用餐者:cyyh
用餐账号ID: 20220014
用餐时间:18:00
用餐地点:(4,6)
菜肴名称:酸菜鱼
金额:18 元
订单一旦确认,即刻备餐,不允许更改!
确认订单输入:yes
取消订单输入:no
yes
订单已经确认!
请按任意键继续. . .
```

我们使用另外一个账号登录到顾客系统中。

我们选中当前时间有人坐的位置。

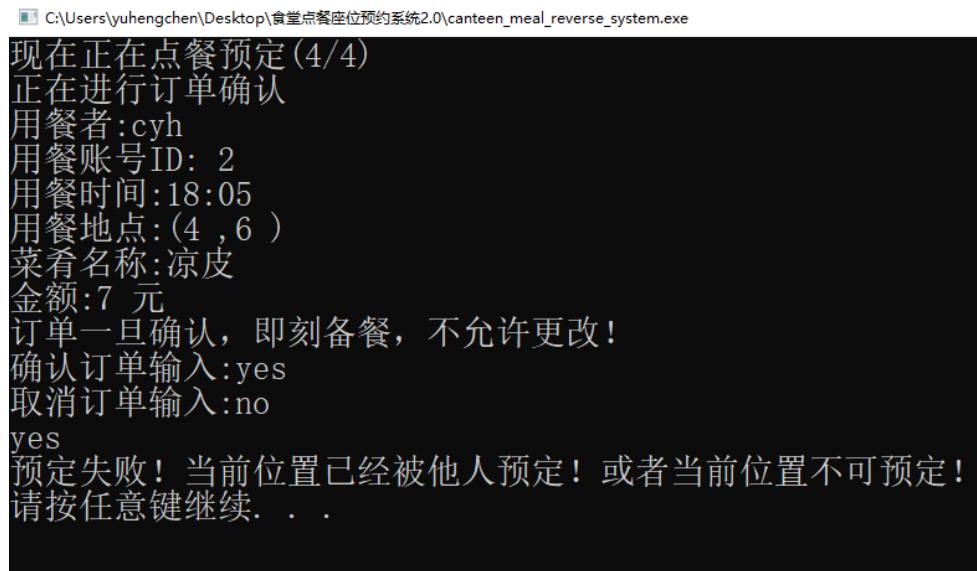
```
现在正在点餐座位预定(2/4)
*****
* '#'代表目前的障碍物或者桌子等,是不可以坐人的位置。 *
* '0'代表目前的座位或者椅子,是可以坐人的位置。 *
*****
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 0 0 0 0 0 0
4 0 0 # # # 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 1 0 0 0 0 0 0 0 0 0
7 0 0 # # 0 0 0 # # # 0 0 0
8 0 0 # # 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请输入坐标,选择座位:
x: 4
y: 6
请按任意键继续. . .
```

确认订单时候会调用check () 函数查看权限。

这个check () 是在写入前一刻调用的，即便两人同时写入最终也只能有一个能写入。

```
bool customer::check(string hour,string min,string sitx,string sity)
{
    //while(!)
    time_log_to_map(hour,min);
    init_map();
    while(!now_map.empty())
    {
        PII t = now_map.back(); now_map.pop_back();
        is_set[t.first][t.second] = true;
    }
    istringstream xxx(sitx);
    istringstream yyy(sity);
    int x_int; xxx >> x_int;
    int y_int; yyy >> y_int;
    return is_set[x_int][y_int];
}
```

在订单确认的时候，会提示无法进行预定。



```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
现在正在点餐预定(4/4)
正在进行订单确认
用餐者:cyh
用餐账号ID: 2
用餐时间:18:05
用餐地点:(4 , 6 )
菜肴名称:凉皮
金额:7 元
订单一旦确认，即刻备餐，不允许更改！
确认订单输入:yes
取消订单输入:no
yes
预定失败！当前位置已经被他人预定！或者当前位置不可预定！
请按任意键继续. . .
```

我们选择有桌子的位置。

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
现在正在点餐座位预定 (2/4)
*****
* '#' 代表目前的障碍物或者桌子等, 是不可以坐人的位置。 *
* '0' 代表目前的座位或者椅子, 是可以坐人的位置。 *
*****
0 1 2 3 4 5 6 7 8 9 10 11 12 13
1 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 # # # 0 0 0 0 0 0 0 0
4 0 0 # # # 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 0 0 1 0 0 0 0 0 0 0 0 0
7 0 0 # # 0 0 0 # # # 0 0 0
8 0 0 # # 0 0 0 # # # 0 0 0
9 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 0 0 0 0 0 0 0
12 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 0 0 0 0 0 0 0 0 0 0 0
请输入坐标, 选择座位:
x: 4
y: 3
```

订单确认时候, 也会提示是无法预定的。

```
C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe
现在正在点餐预定 (4/4)
正在进行订单确认
用餐者:cyh
用餐账号ID: 2
用餐时间:18:03
用餐地点:(4 , 3 )
菜肴名称:酸菜鱼
金额:18 元
订单一旦确认, 即刻备餐, 不允许更改!
确认订单输入:yes
取消订单输入:no
yes
预定失败! 当前位置已经被他人预定! 或者当前位置不可预定!
请按任意键继续. . .
```

打开当日的文件。根据用户名索引订单。订单中展示了订单目前的状态。

```
1:
用餐时间:18:00
用餐地点:(4,6)
菜肴名称:酸菜鱼
金额:18 元
订单状态:餐肴还在制作中, 请等待!

请按任意键继续. . .
```

索引代码如下

```
void customer::show_the_log()
{
    ifstream read;
    string log;
    read.open(reserve_log_txt_name, ios::in);
    int cnt = 1;
    while(getline(read, log))
    {
        if(log_to_item(log, 7) == get_name())
        {
            cout << cnt << ":" << endl; cnt++;
            cout << "用餐时间:" << log_to_item(log, 1) << ":" << log_to_item(log, 2) << endl;
            cout << "用餐地点:" << "(" << log_to_item(log, 4) << "," << log_to_item(log, 3) << ")" << endl;
            cout << "菜肴名称:" << log_to_item(log, 5) << endl;
            cout << "金额:" << log_to_item(log, 6) << "元" << endl;
            if(log_to_item(log, 10) == "0") cout << "订单状态:餐肴还在制作中, 请等待!" << endl;
            else cout << "订单状态:餐肴已经制作完成!" << endl;

            cout << endl;
            //system("pause");
        }
    }
}
```

6.2.7 商家系统

6.2.7.1 商家界面

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
#####  
南京信息工程大学 自助点餐预定系统欢迎您! #  
#  
Welcome to canteen_meal_reserve_system of NUIST! #  
#####  
当前的时间是: #  
2022-12-23 16:42:01 #  
#####  
这里是商家客户端  
#####  
wj, 欢迎您!  
#####  
输入数字, 进行操作:  
0:退出系统  
1:查看菜单  
2:增加菜品  
3:删除菜品  
4:查看需要处理的订单  
5:处理订单
```

6.2.7.2 菜谱的增删改查

进入菜品增加功能后,输入菜品, 和价格, 即可增加成功。

同样系统会自动分配条目id。

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

```
请输入菜品:  
红烧牛柳  
请输入价格:  
17  
增加菜品成功!  
请按任意键继续. . .
```

我们使用菜谱查看功能查看。

菜品是写入menu.txt的。

查看菜谱接口是user基类的功能。

所以在商家和顾客调用的菜谱是同一个菜谱。

但是只有商家有修改菜谱的功能。

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_sysi

序号	菜肴名称	价格（元）
2	红烧牛肉面	15
3	酸菜鱼	18
5	青菜炒肉	10
6	酸菜肉丝面	8
7	黄焖鸡米饭	22
9	凉皮	7
11	红烧肉	16
12	红烧牛柳	17

请按任意键继续. . .

输入菜肴编号，进行条目删除。

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_syste

序号	菜肴名称	价格（元）
2	红烧牛肉面	15
3	酸菜鱼	18
5	青菜炒肉	10
6	酸菜肉丝面	8
7	黄焖鸡米饭	22
9	凉皮	7
11	红烧肉	16
12	红烧牛柳	17

请输入需要删除的菜品编号

6

删除菜品成功！

请按任意键继续. . .

效果如下

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

序号	菜肴名称	价格（元）
2	红烧牛肉面	15
3	酸菜鱼	18
5	青菜炒肉	10
7	黄焖鸡米饭	22
9	凉皮	7
11	红烧肉	16
12	红烧牛柳	17

请按任意键继续. . .

6.2.8 顾客系统与商家系统通过文件系统的交互

商家进入订单处理功能界面。输入备好餐的编号进行处理。

C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe

以下订单需要处理

订单id: 37
用餐时间:18:00
菜肴名称:酸菜鱼
金额:18 元

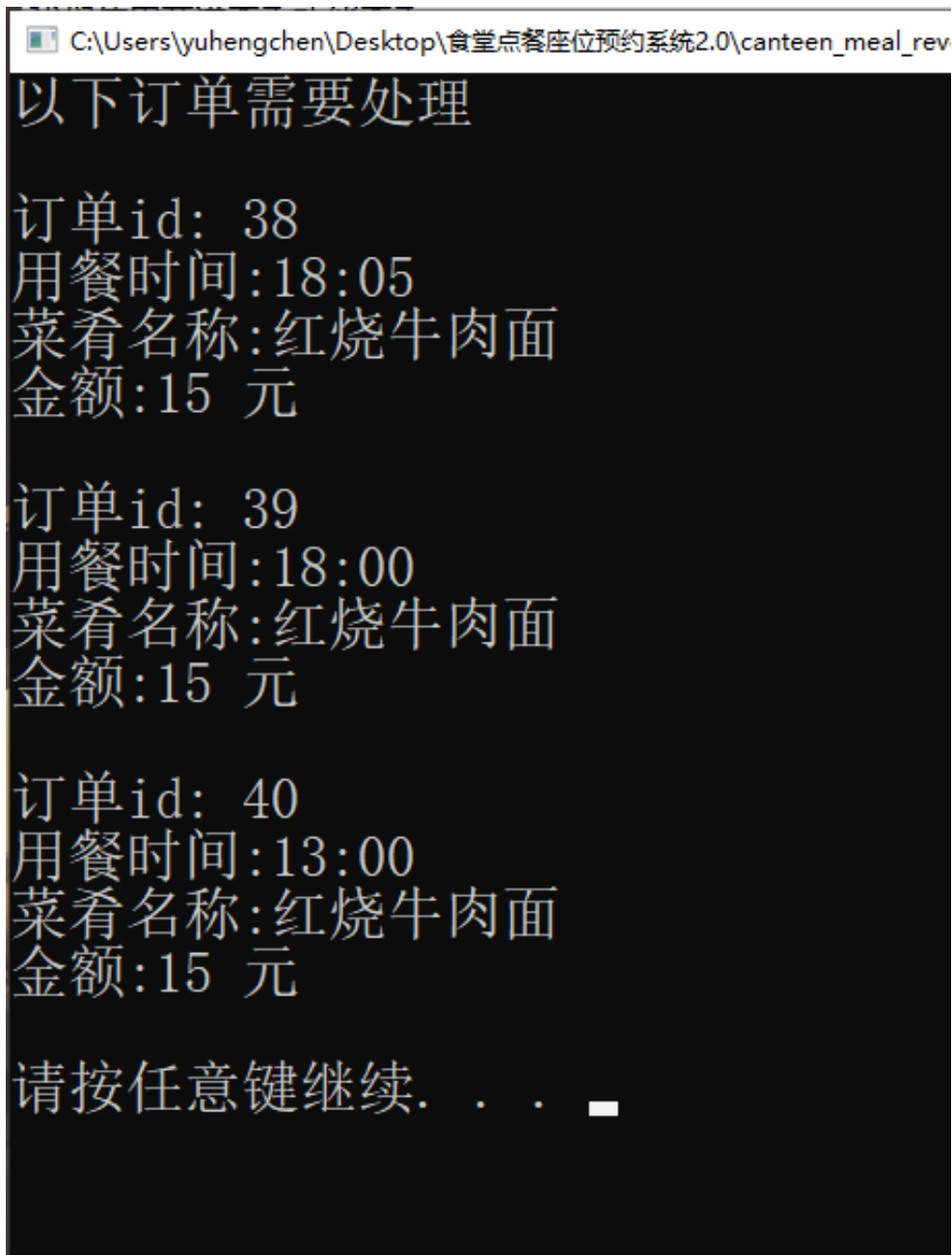
订单id: 38
用餐时间:18:05
菜肴名称:红烧牛肉面
金额:15 元

订单id: 39
用餐时间:18:00
菜肴名称:红烧牛肉面
金额:15 元

订单id: 40
用餐时间:13:00
菜肴名称:红烧牛肉面
金额:15 元

请输入您已备好餐的订单编号:37_

商家使用订单查看功能，可以发现刚刚处理的订单，已经消失。



商家在处理订单的时候。实际上是向订单尾部追加了一个状态量 代码如下

```
void store_keeper :: deal_with_the_log()
{


    show_log_need_to_deal(); cout << endl;
    string id;

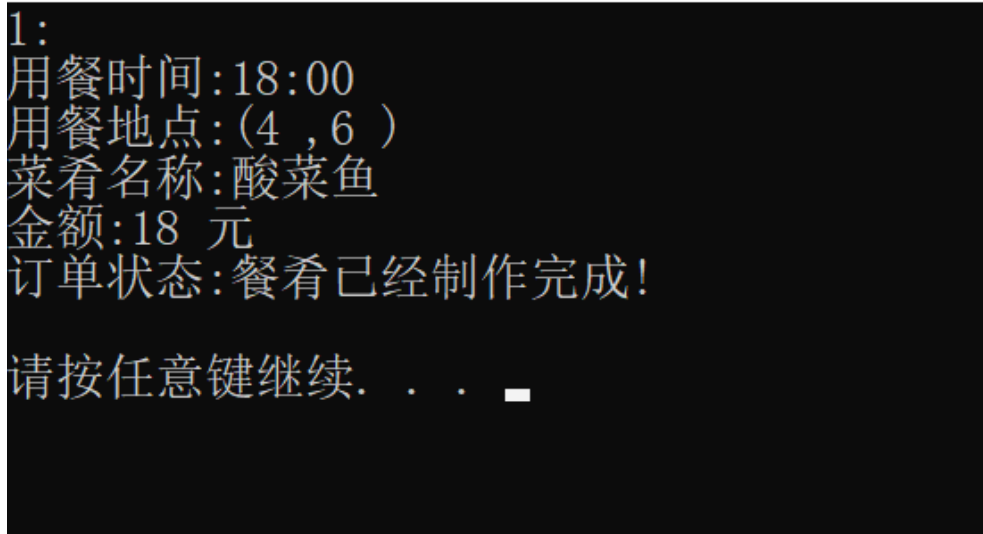
    cout << "请输入您已备好餐的订单编号:" ;cin >> id; id += ' ';
    fstream stream;
    stream.open(reserve_log_txt_name,ios::in);
    queue<string> t;
    string log;
    while(getline(stream,log))
    {
        if(log_to_item(log,9) == id) log += "1";
        t.push(log);
    }
    stream.close();
```

```
stream.open(reserve_log_txt_name,ios::out | ios::trunc);

while(!t.empty())
{
    stream << t.front() << endl;
    t.pop();
}
stream.close();
cout << "已经处理完成!" << endl;
}
```

回到顾客的账号，进行订单状态查看,我们发现，订单状态已经更新为制作完成。

 C:\Users\yuhengchen\Desktop\食堂点餐座位预约系统2.0\canteen_meal_reverse_system.exe



1:
用餐时间:18:00
用餐地点:(4,6)
菜肴名称:酸菜鱼
金额:18 元
订单状态:菜肴已经制作完成!
请按任意键继续. . .

7 总结与思考

7.1 项目总结

对于整个项目来说，完全是自己一个人独立完成，做了下来，很有成就感。整个项目，功能齐全。可用性高。项目，代码模块化很好，在变量的命名上，下了功夫，代码可读性强，也便于后期的维护。在类的使用上，熟练的使用了类的继承。减少了代码的冗余。但是鉴于时间和精力，还有能力，有些好的想法暂时还没有实现。目前，该软件是2.0版本。相较于之前自己的1.0版本，革新是，增加了reserve_log文件夹来记录每天的订单数据。这在我看来是跃升的一大步，且很有必要。

7.2 遇到的错误和改进

- (1)向文件里写数据的时候，先写了换行，再写内容。导致txt文本里面多了很多空行。再从文件里读数据，读到空行，然后使用substr解析的时候，报错。反复调试很久，才发现这个问题。然后改成先写内容再写换行，才解决了问题。
- (2)写登录系统的时候，忘记给为找到账号那条分支，增加计数操作，导致程序调试一直卡在找账号那里。
- (3)初始化地图的时候忘记使用memset () 清除前置操作，导致，座位信息一直被前几步干扰。后来整改了。

7.3 有所收获

- (1)不得不提，1.5k行 代码，锻炼了打字速度，很爽。
- (2)熟悉了STL库里的string类，vector类，queue类等等。尤其是string类的substr()，使用他发明了一个解析string的轮子。
- (3)熟悉了文件的输入输出流，在文件级别的操作收获颇丰。尤其学习到了，在cpp里写相对路径。
- (4)成就感，边写边查，学到了这种在开发中学习的学习方法。
- (5)学习到了一些dos命令如 "cls"等等
- (6)学到了cpp怎么调用时间，对以后开发一些cpp小游戏，可能会有所帮助
- (7)写文档的时候还学习到了markdown语法，css样式，chorm插件的相关知识。
- (8)学到了如何使密码不回显。
- (9)项目的整体架构把握能力提升。

7.4展望

以下是希望后期可以实现的。

- (1)修改密码模块没有写，后期添置。
- (2)给手机发送验证码功能和支付宝支付验证功能，可能要使用到云服务器的接口，后期学习。
- (3)如何使不同商家管理自己的菜单，然后用户调用的是整体的菜单。这是现有的知识可以实现的。大概在文件名 级别上进行操作。后期添置。
- (4)管理员的账号的思考。也应该是可以修改密码的。然后，有一个root管理员。然后，只有root管理员可以 创建管理员用户。所有管理员对于地图的操作应当在后台有记录。防止管理员不按规定篡改地图。可以后期问责。
- (5)地图模块。在文件名级别上操作。可以有很多的食堂。这就要记录,商家属于哪个食堂。订单也要记录到食堂和商家。
- (6)座位模块，以后需要增加外带功能，也就是说，学生是可以带回宿舍的。
- (7)扫码点单功能。这个功能已经很多软件都实现了。
- (8)软件的越界检查也需要后期完善。
- (9)密码的密文存储，后期也希望实现。

8 参考文献

- [1] 郑莉 董渊 何江舟.《C++语言程序设计》.清华大学出版社, 2010
- [2] [美] Thomas H. Cormen.《算法导论》.机械工业出版社, 2012
- [3] [美] Stephen Prata.《C++ Primer Plus》.人民邮电出版社, 2020
- [4] 严蔚敏 吴伟明.《数据结构c语言版》.清华大学出版社, 2007
- [5] 赵驰龙 程努华 姜晔.《使用软件工程》.电子工业出版社, 2020