

**《程序设计课程设计》实验二**

**实验报告**

题 目： 《远程控制贪吃蛇食物大逃杀》

学 院： 数理与电子信息工程学院

专 业： 网络工程

班 级： 17网工一班

学 号： 17211134120

姓 名： 吴佳帅

指导老师： 李忠月

完成日期： 2018年3月

**目录**

[第1章 项目介绍 1](#_Toc478661865)

[1.1 游戏介绍 1](#_Toc478661866)

[1.2 亮点介绍 1](#_Toc478661867)

[1.3 难点介绍 1](#_Toc478661868)

[第2章 系统设计 1](#_Toc478661869)

[第3章 关键技术介绍 1](#_Toc478661870)

[第4章 软件说明书 1](#_Toc478661871)

[第5章 总结 2](#_Toc478661872)

[参考资料 2](#_Toc478661873)

# 第1章 项目介绍

## 1.1 游戏介绍

这个游戏是属于双人远程控制游戏，双人指的是一个人控制贪吃蛇追杀食物，一个人控制食物逃离贪吃蛇。其中贪吃蛇的按钮控制为“w,a,s,d”其中w是向前运动，a是向左运动，s是后退，d是向右边运动。食物的按钮控制为“i,j,j,k,l”其中i是向前运动，j是向左运动，k是后退，l是向右边运动。游戏规则是：贪吃蛇不能碰到墙壁。贪吃蛇吃到食物加十分。食物必须逃离贪吃蛇，期间不能触碰贪吃蛇，并且不能触碰墙壁。

## 1.2 亮点介绍

亮点有两个，一个是双人游戏，另外一个是远程投影，及在一台电脑上展示的贪吃蛇会完整的映射到另外一台电脑上面。

## 1.3 难点介绍

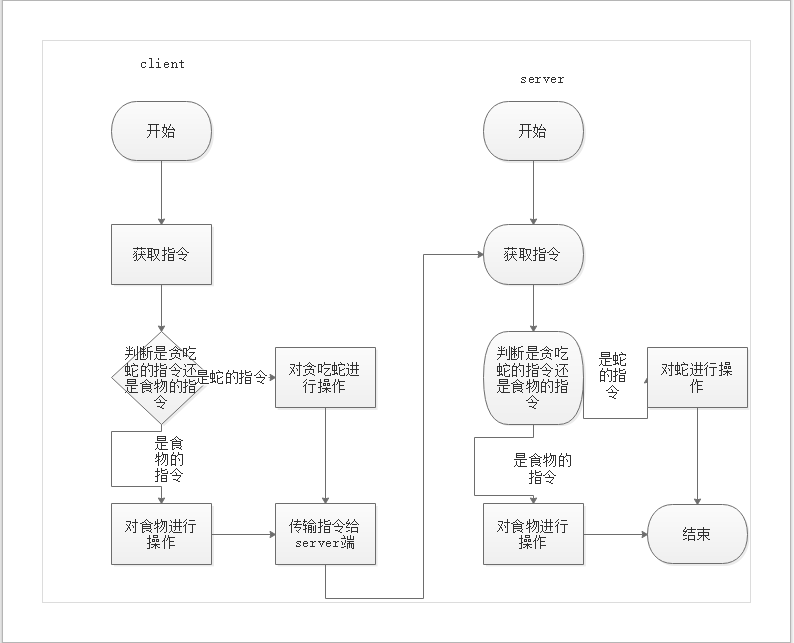
一共有以下几个难点：

* 如何将程序的通讯部分和贪吃蛇部分合起来。
* 蛇和食物的交互性问题；
* 程序的通讯问题；
* 如何实现蛇的动态运动
* 如何实现初始化贪吃蛇
* 实现颜色的改变
* 蛇如何实现自身长度的增长
* 判断蛇与食物之间的交互关系
* 如何分离蛇与食物的指令交互问题
* 如何实现蛇运动的速度的改变

这些难点会在下面的关键技术这里解决。

# 第2章 系统设计

这里分为两个系统第一个系统是贪吃蛇系统，第二个系统是网络编程udp系统



图一

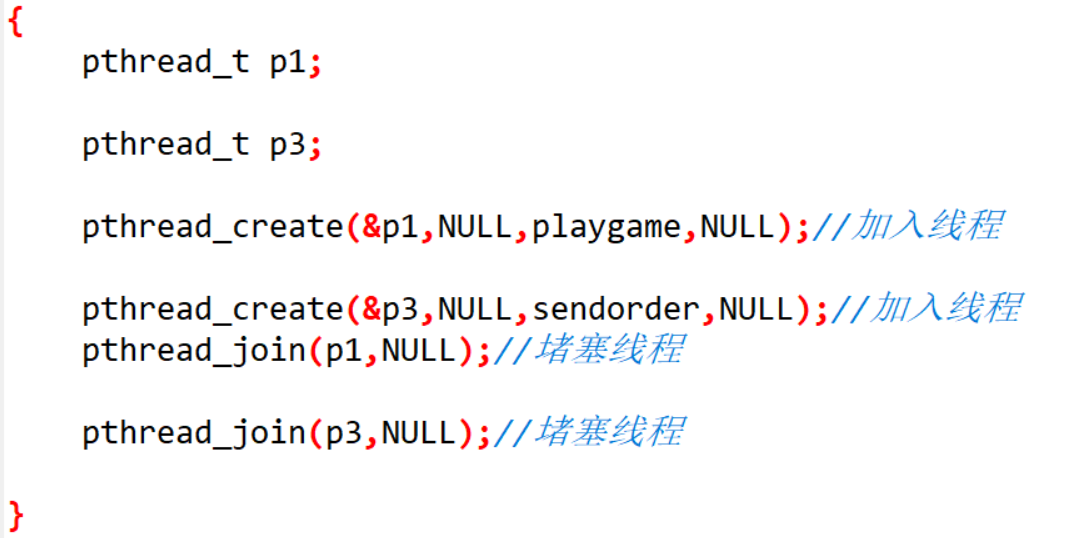
图一展示的就是这个系统。其中客户端接受指令发送指令到server端，可以使得server和client同时执行相同的指令，从而达到实现远程控制的目的。

# 第3章 关键技术介绍

如何将程序的通讯部分和贪吃蛇部分合起来？  
这里通过线程实现将两个程序结合在一起(我本然想通过多进程实现两部分的结合，但是发现每个进程之间相同的变量的内容是不同的，相当于还是执行两个程序，所以采用了多线程)

多线程如何实现？

建立线程，堵塞线程达到目的。



图二

图二展示了线程在C语言中的实现，首先定义线程指针，用来指向对应的线程，接着将函数加入线程，最后堵塞线程。

### 蛇和食物的交互性问题；

分为：

蛇和食物是否碰撞

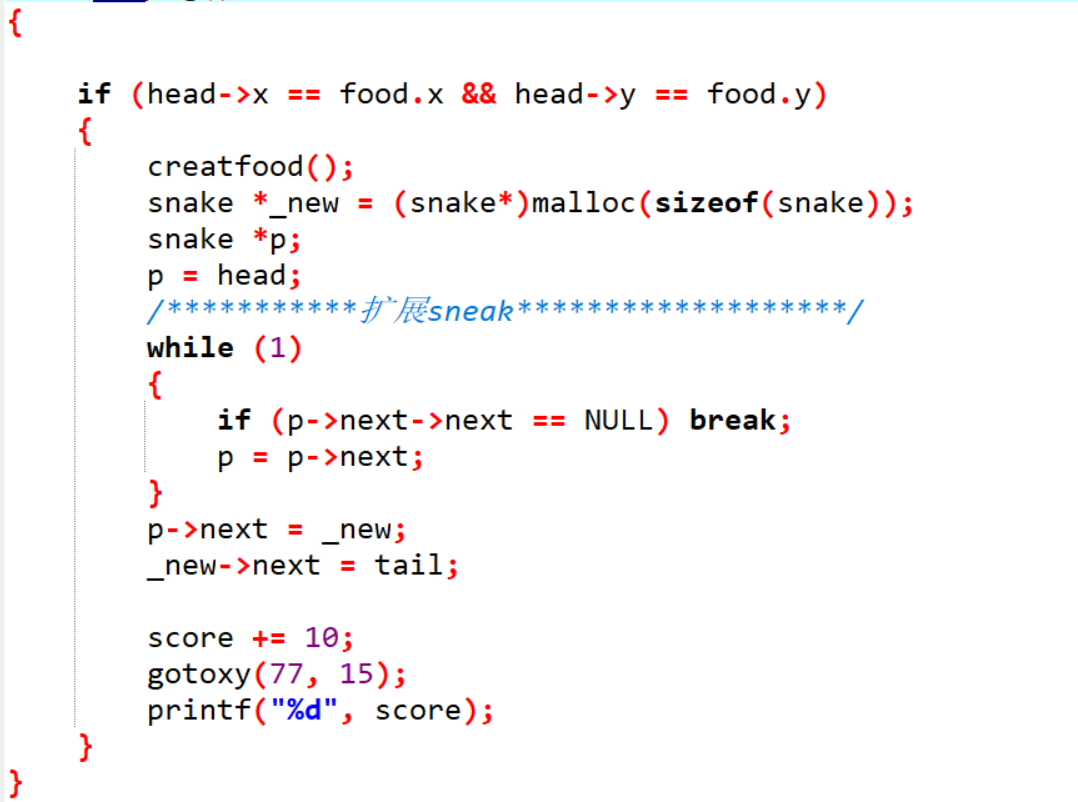
对蛇进行一次遍历来比较蛇的x,y坐标是否与食物的x,y坐标相等



图四

图四就是对蛇身进行一次遍历，判断蛇与食物的坐标是否相同

蛇是否吃到食物



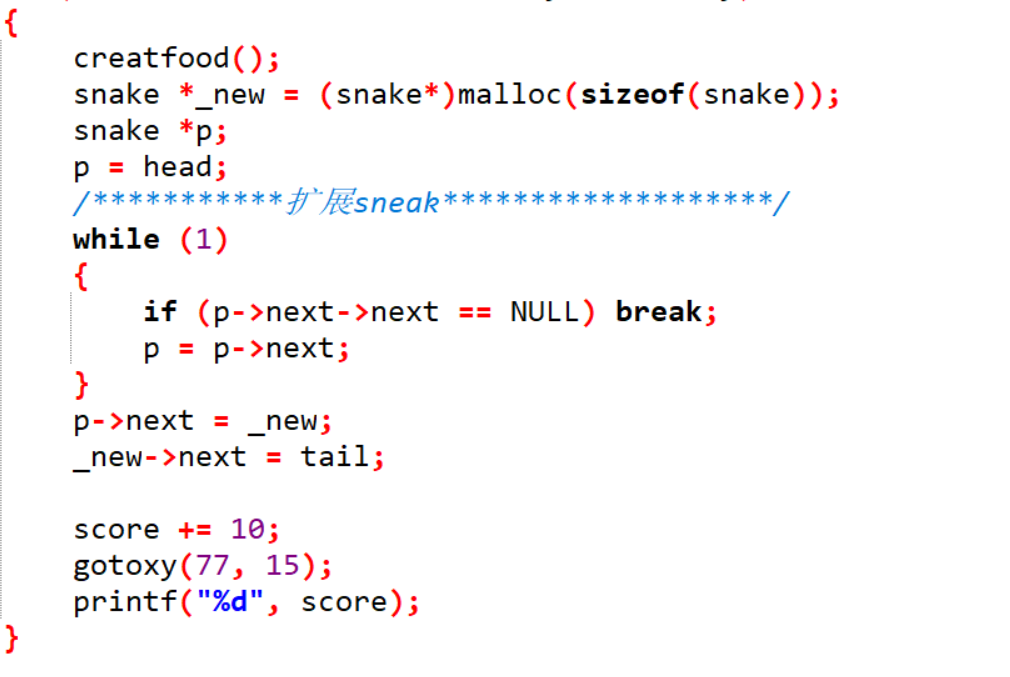
图五

就是判断蛇的头部的坐标是否与食物的坐标相同

分离指令，将食物的指令和贪吃蛇的指令分离开来，我用的是strchr这个函数去判断指令是不是属于那八个指令，指令又分别是属于哪个指令。



### 蛇如何实现自身的增长



图六

先创造一个新的节点，再将每一个节点，向前移动，最后再将最后一个节点和新的节点连接再一起就可以了。

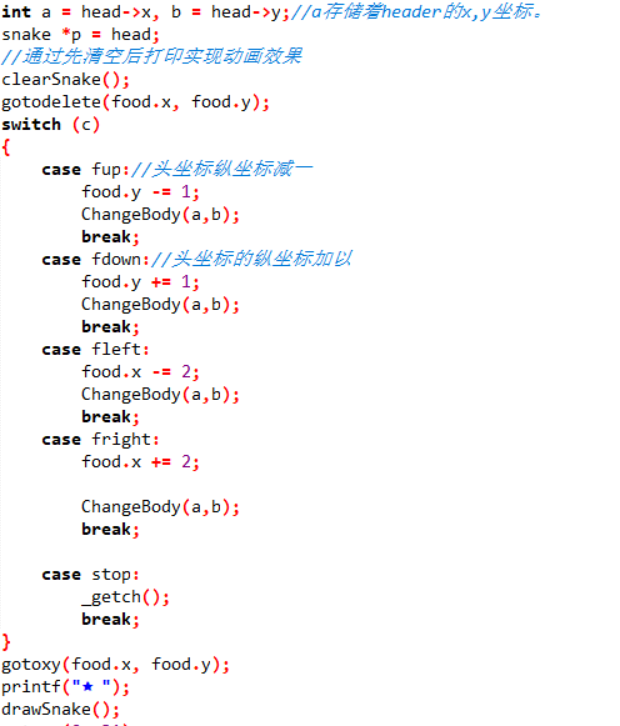
### 蛇的指令如何去实现



图七

及先对蛇的头部进行坐标的移动，再移动整个蛇像的移动再进行删除蛇的画像，再将新的画像打印上去，

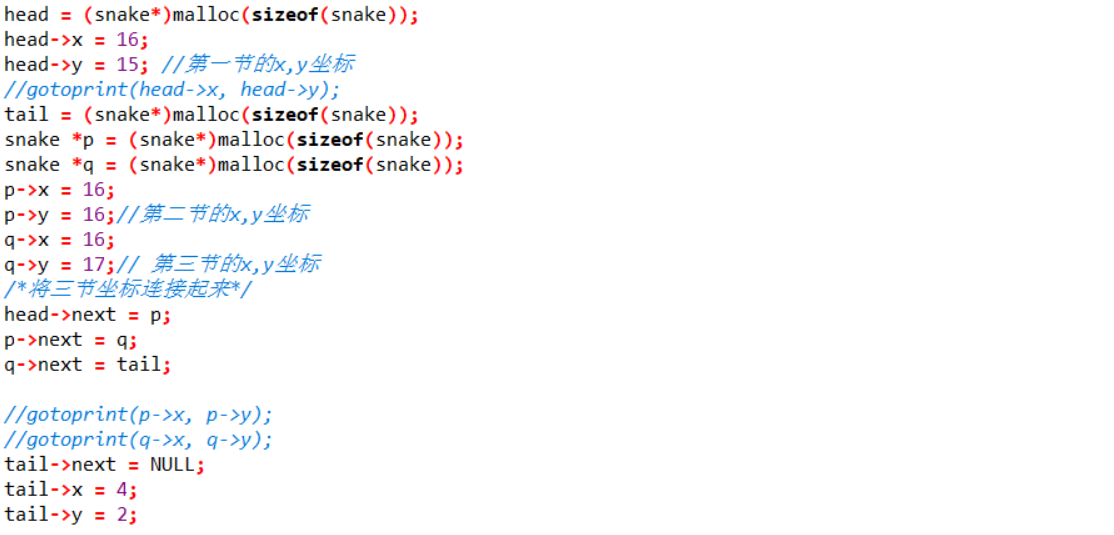
其中要注意的是由于打印的是中文，所以横坐标每次都要加2,纵坐标每次都是加1.

如何实现食物的移动：  


图八

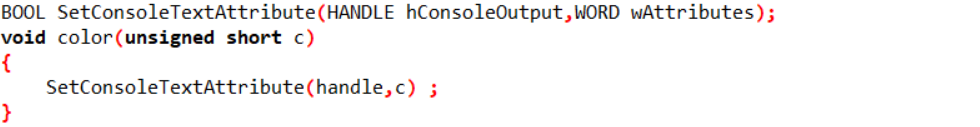
图八显示的就是食物对指令的反应，已经如何实现食物的动画，食物对指令的反应与蛇队指令的反应基本一样，这里不一样的是，重新打印不仅仅需要打印食物还要把蛇的图像一起打印一遍。

### 初始化贪吃蛇



### 实现窗口颜色的改变

首先获取窗口的句柄，其次，移动光标并改变颜色

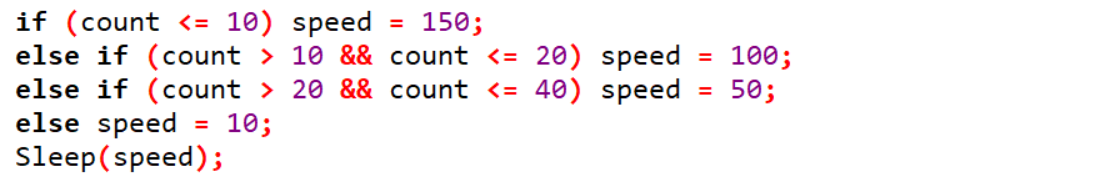


### 如何实现蛇的动态运动

只要让蛇在运行指令后暂停不同的时间点即可。

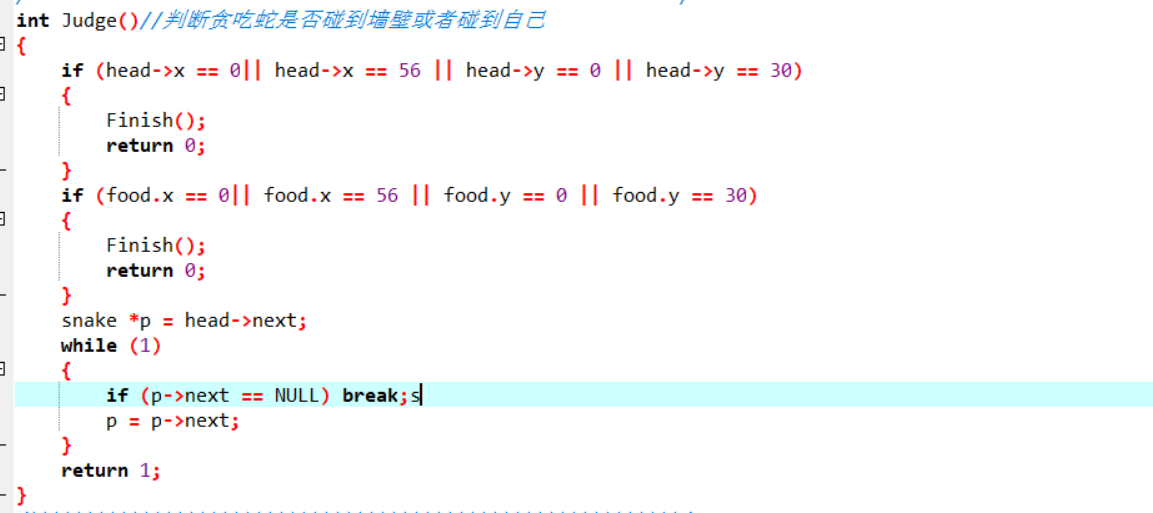
### 如何实现蛇的速度的改变

只要改变根据蛇的长度改变不同的暂停时间即可。



count是一个全局变量来记录蛇的长度。

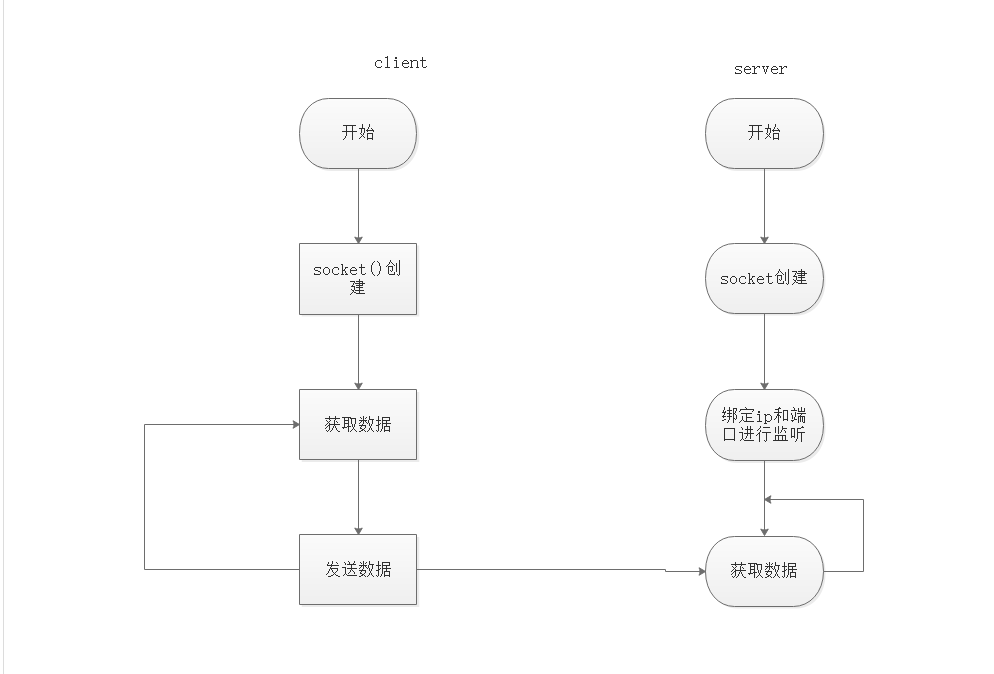
### 如何判断蛇的死亡



### 如何实现贪吃蛇的远程控制：

上面主要是讲贪吃蛇部分的难点，下面来讲下网络通讯的难点。

首先先放一张贪吃蛇网络通讯的图

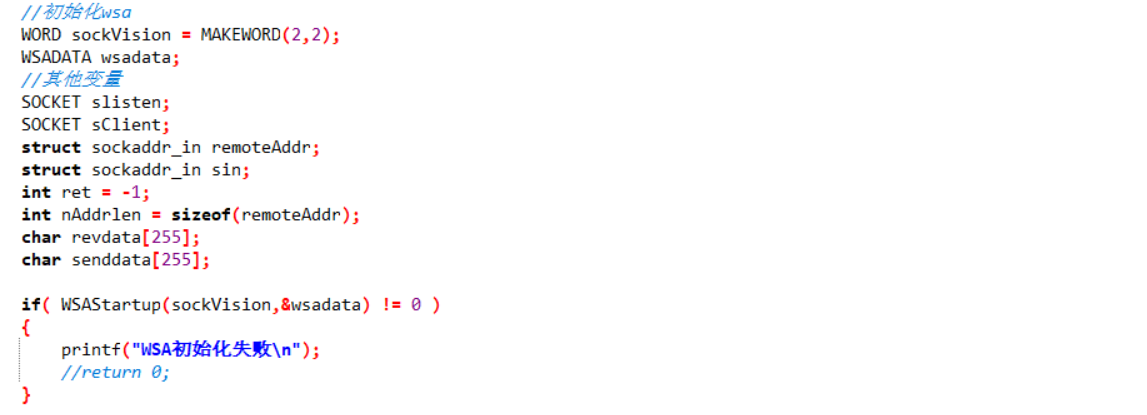


图九

图九展示的就是贪吃蛇的指令传输流程图。我将蛇的指令和食物的指令分别存在两个全局变量里面sclick和fclick因此只要改变了这两个变量的话，就能够实现远程控制（因为每次执行的命令都是从这两个命令变量里面拿到的。）

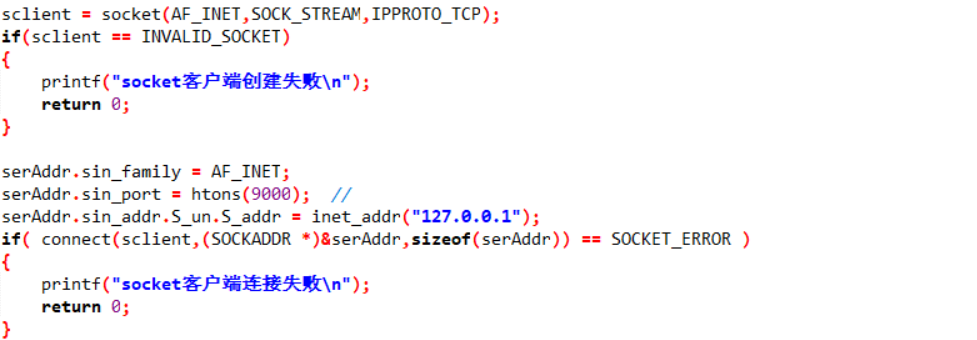
我们现在来看client端也就是发送指令的这一边。

首先创建socket，初始化各种变量，放入要发送的ip地址和接收的端口号。



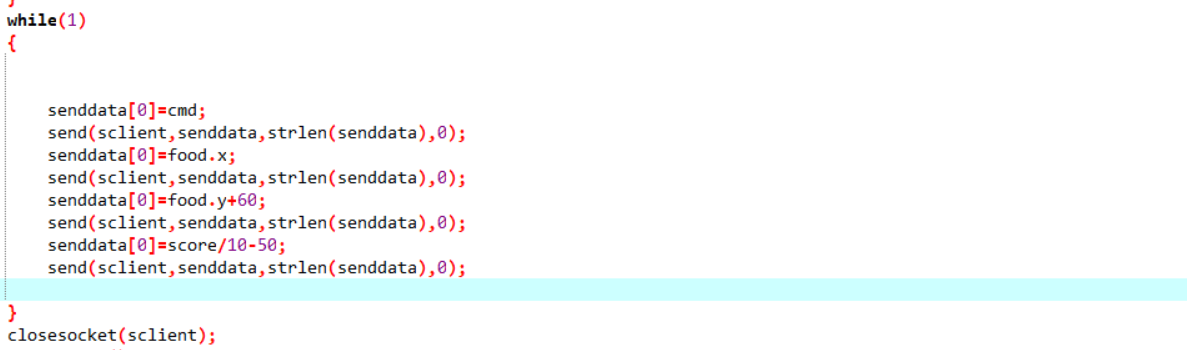
图十

图十表示的就是初始化



图十一

图十一表示的是创建socket客户端绑定要发送的ip地址和端口号，并且尝试第一次连接。

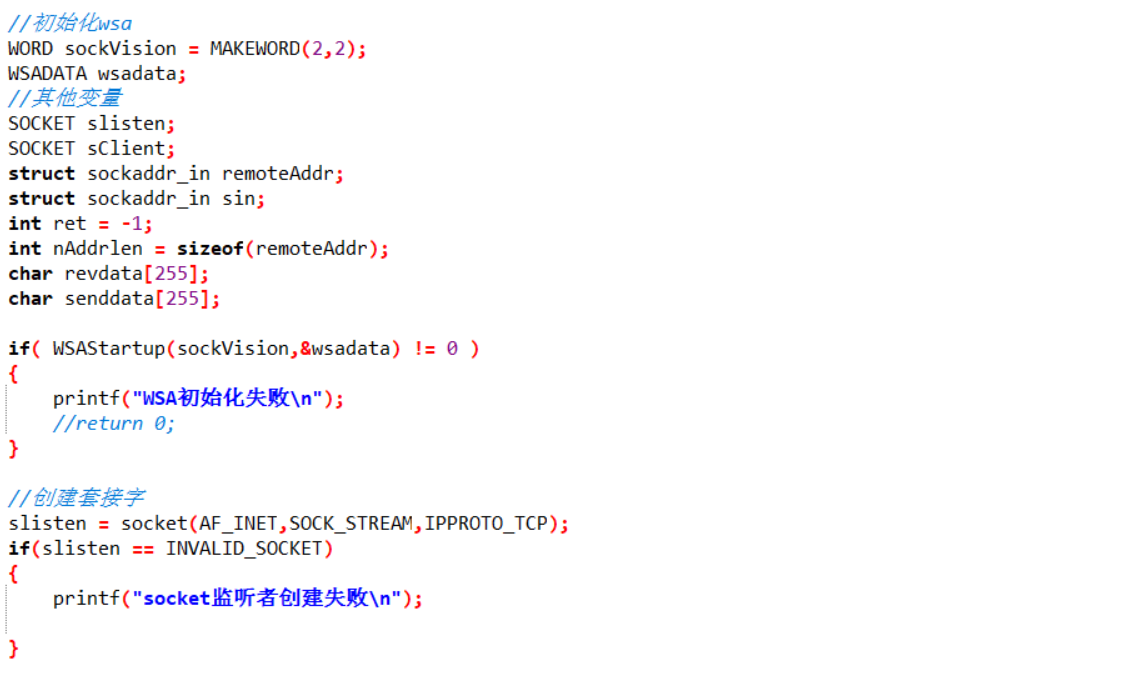


图十二

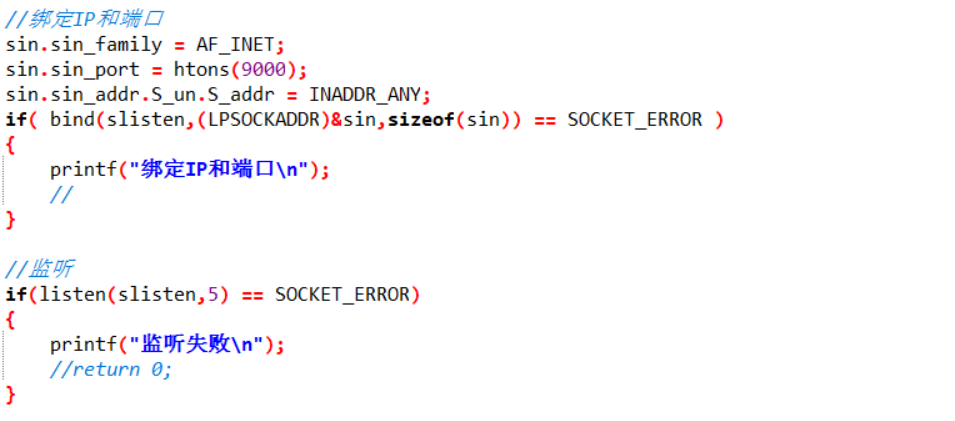
图十二表示的是发送数据，这里为什么要发送的这么麻烦，我们等讲到server这里再说。

server端

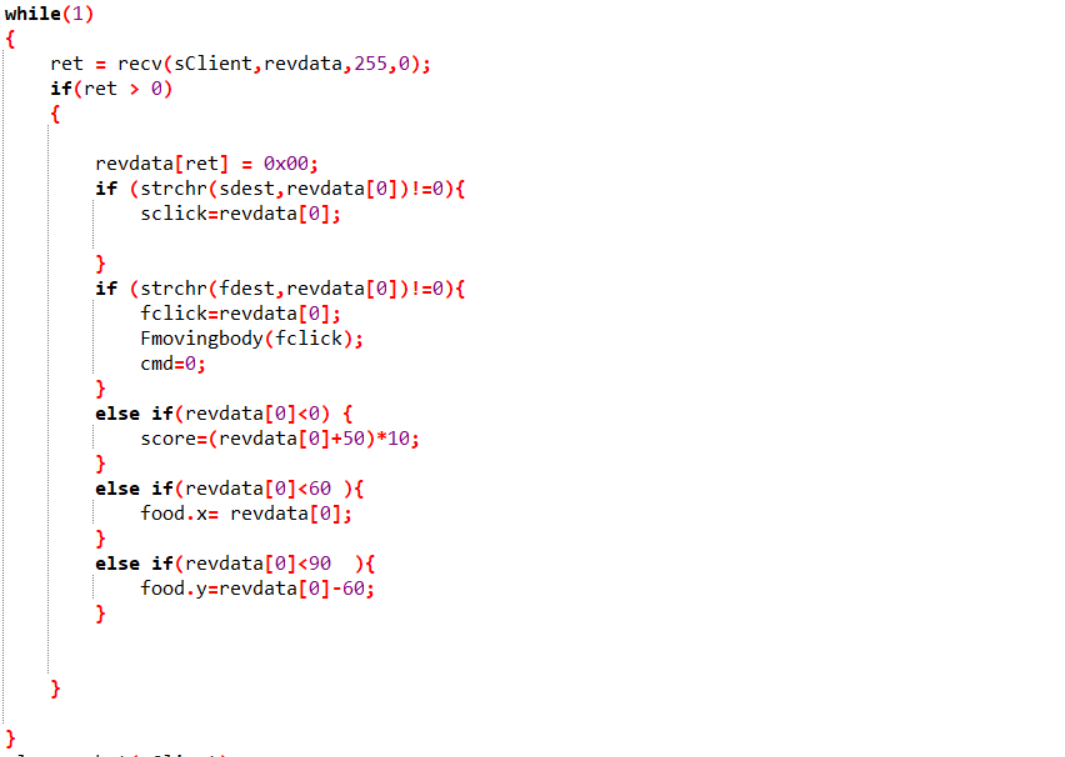
一样都是要先进行初始化，不过这里绑定的是获取到数据的端口号。

初始化：  


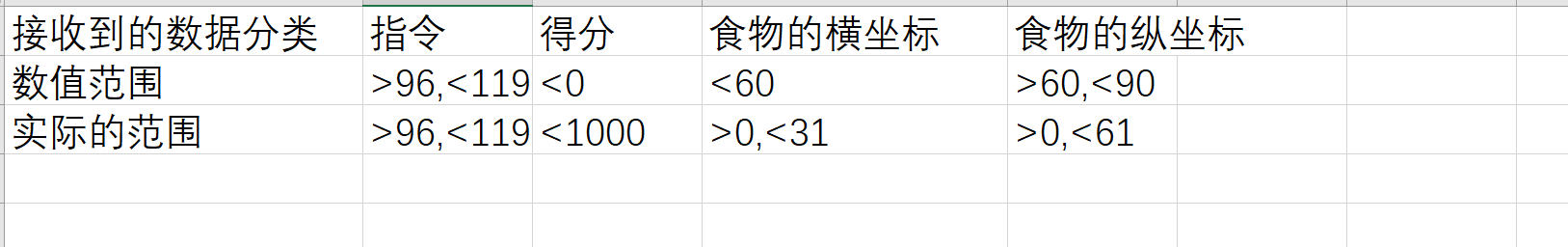
图十三

绑定：  


图十四

获取数据：  


这里来讲下为什么要收发数据真么麻烦，因为send（）函数只能进行发送字符指针，因此只能用assic码的方式来分割需要接收到的数据。



因此我采用的是用assic码的方式来区分指令。

# 第4章 软件说明书

连接：运行food5.cpp几秒后,再运行snack.cpp如果food上显示出食物，表示连接成功。

指令：八个指令再游戏介绍出已经说明。

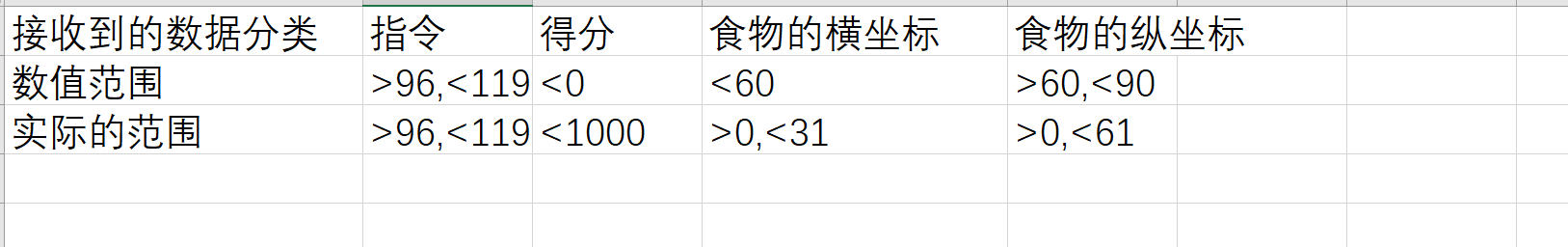
# 第5章 总结

实验过程中遇到的问题：

1. 无法将贪吃蛇部分的代码和网络编程那部分的代码合起来，本然是想用多进程，发送用了之后，并不能进行指令的相通，后来才知道多进程的每个进程之间是不相通的就相当于两个独立的程序，虽然是在同一个文件中运行。
2. 贪吃蛇部分：无法使得向后退不死，本然再贪吃蛇指令的那块改动了三次。

第一次就是直接暂停，发现根本不能运行，第二次是又定义了一个变量先将这个变量的坐标进行改变，在和第二节贪吃蛇进行改变，发现还是不行，运行的时候，发现第一节和第三节断掉了。第三次读了后面judge()函数那块，把碰到自己的身体就死去的注释掉才解决这个问题。

1. 网络通讯这块：传输指令这块弄得比较久，本然我是想将python和C语言一次来搞，但发现不知道为什么不能成功，因为指令只能是按照字符得数据类型进行传输，我用了整数类型，用了自定义结构类型，用了…最后采用了assic码进行分段，因为刚好我要传输得数据得assic都属于不同得几个区间。



未解决的问题：

server接收到指令后，移动食物的时候，还是会出现贪吃蛇画的会失败，会出现好多个食物的图案.

实验的心得与体会

通过这次实验我才发现原来什么事情都要去做一下，虽然磕磕绊绊，但是最后还是成功了一般，为什么是一半，因为我本然想做的是贪吃蛇联机双人小游戏，可能时间不够，所以就只能降低了要求，而且改动其他人的代码，就必须要对其他人的代码有个仔细的了解。

# 

# 参考资料

1. 简易csdn贪吃蛇
2. [windows下简单的C语言服务器（TCP）](https://blog.csdn.net/mastic_29/article/details/70245038)
3. [C语言的多线程](https://blog.csdn.net/hxpjava1/article/details/54375547)