# 游戏说明

游戏名称：Gluttonous Snake

游戏平台：PC(WINDOWS)

开发软件：win10，vs2010

作者：莫廷钰（自1710）

## 游戏简介

**玩家通过控制蛇的移动来获取地图上随机产生的食物，在蛇移动的过程中，蛇的生命值随着蛇的生存时间增加而下降，蛇可以通过获取食物来增加生命值，蛇的生存时间越长分数越高。与此同时，蛇在移动的过程中不能咬到自身或撞上墙壁，否则游戏结束**。

## 操作说明

**主界面：**

**(1)PLAY (玩家按下数字键1进入游戏界面开始游戏)**

**(2)Instructions (玩家按下数字键2进入说明界面看游戏操作说明)**

**(3)Level Ajustment（玩家按下数字键3进入游戏难度等级更改界面以改变游戏的难度）**

**(4)EXIT (玩家按下数字键4退出游戏)**

**游戏界面：**

**控制蛇身移动:W（蛇头向上移动）S(蛇头向下移动) A(蛇头向左移动)D (蛇头向右移动)**

**SPACE(空格 按下使得游戏中途退出)**

## 3. 流程（主函数）设计

**main()->Interface()(主界面函数)->process->永循环switch分支结构-(根据玩家键入的选择进行操作)**

**->move()(玩家选择PLAY)**

**or ->Instructions()(玩家需查看说明)**

**or->level\_ajustment()(玩家调整游戏难度)**

**or->EXIT()(退出游戏)**

## 4. 功能实现（其他函数）设计

**void move();**

**蛇的移动函数。内部语句大致为循环结构和分支结构，循环结构有蛇体跟踪蛇首循环（若使蛇移动则只需控制蛇头，蛇其余部分只需模仿前一部分的运动）和蛇体打印循环（通过光标的移动打印蛇身，蛇尾依据捕食与否打印空格或蛇体），分支结构有switch开关分支结构（用于控制蛇头的移动方向），除去上述语句，该函数内部还设置了分数增长并输出，生命值衰退并输出的语句**

**void delay(double delay\_time);**

**延时函数。以所需延续的时间为实参，进行程序的延时进行**

**void hide\_CURSOR();隐藏光标函数。**

**void init();开始游戏界面初始化函数，生成墙体和蛇体和食物的初始位置**

**void Snake\_inflate();蛇的长大函数。**

**蛇的本质为结构体数组指针，每一个数组元素为该蛇的部分所定位的横纵坐标，当蛇头坐标与食物坐标重合，调用该函数，新动态内存分配一个比原蛇长度大1的结构体数组，然后将原蛇的坐标从新数组尾部开始复制，最后将原蛇的尾部相对前一个坐标赋值给新数组的首个元素，这样使得新蛇形成，然后定义一个相同类型的工作指针，将原蛇的指针赋值给工作指针，之后再将新蛇的指针值赋给原蛇，最后用工作指针将原蛇原先申请的内存释放。最终完成蛇的长大。**

**void genarate\_MEAT();食物生产函数。在random()函数产生的坐标打印食物**

**void random();随机数函数。**

**生产食物，若食物在墙外或蛇身上，重新调用该函数生成新的食物坐标。**

**void check();检查游戏是否继续进行函数。**

**内部大多语句为if判断语句，判断蛇是否吃到食物，蛇是否自咬，蛇是否撞墙。**

**void gotoxy(int x, int y);光标移动函数。**

**将光标移动到指定位置，从而进行打印工作，该游戏制作的核心，若用system(“cls”)来刷新输出即表示蛇的移动，会导致整个游戏界面抖动。故用该函数改良，墙体不需要打印多次。而蛇体的打印，只需要用一个循环结构将光标移动至结构体数组中储存的坐标，然后打印蛇体，尾部打印空格以达到清除效果。利用外部循环嵌套控制蛇头的持续移动，从而达到蛇的整体移动效果。**

**void spy(char \*object);键入监测函数.**

**Object为键入的对象，该函数监测玩家是否键入，若键入则将该值赋给Object.**

**void gameover();游戏结束函数.**

**游戏结束界面和一些变量的赋值等收尾工作**

**void Interface();主界面函数。**

**人机交互的平台，根据主界面的提示可以达到交互的效果，理解编程人的想要用户完成的操作**

**void PLAY();游戏开始函数。**

**void EXIT();关闭游戏函数。将蛇所占用的内存释放后结束程序。**

**void Wall();生成围墙函数。**

**void Level\_ajustment();游戏难度等级调整函数。**

**进入该函数会有一个调整界面，输出玩家可选择的等级，当玩家按下相应的等级序号，程序会有一段游戏难度展示，让玩家了解实际情况并做出自己的选择。**

**void Demonstrate(double Speed\_level);游戏难度展示函数。**

**void process();主过程。**

## 5. 工程文件包

见附件。

## 6. 组员分工 单人完成。