**《Python程序设计》**

**课程设计报告**

　　　　　　　　　　（2018 —2019 学年第 一 学期）

题 目 贪吃蛇游戏设计

学生姓名 黄帅哲

专业班级 信管16-2

学生学号 311609030219

教师姓名 徐 文 鹏

成 绩**：**

评 语：

教师签名：

日期：

目录

[1 设计目的 3](#_Toc535310759)

[1.1课程设计目的 3](#_Toc535310760)

[1.2 题目选择： 3](#_Toc535310761)

[1.3 设计来源 3](#_Toc535310762)

[2 课程设计和要求： 3](#_Toc535310763)

[2.1 课程设计教学任务和要求 3](#_Toc535310764)

[2.2课程设计具体要求 4](#_Toc535310765)

[3总体设计 4](#_Toc535310766)

[3.1 小组人员分配 4](#_Toc535310767)

[3.2 任务分配： 5](#_Toc535310768)

[3.3具体功能设计： 5](#_Toc535310769)

[4 设计实现 7](#_Toc535310770)

[4.1 程序运行效果 7](#_Toc535310771)

[4.2 实现方法 7](#_Toc535310772)

[4.2.1 开始和结束界面的编辑 8](#_Toc535310773)

[4.2.2 键盘控制的实现 8](#_Toc535310774)

[4.3实现结果评价 9](#_Toc535310775)

[5详细设计 9](#_Toc535310776)

[5.1界面代码 9](#_Toc535310777)

[5.2详细介绍 10](#_Toc535310778)

[5.2.1提示信息函数介绍 11](#_Toc535310779)

[5.2.2开始界面函数介绍 11](#_Toc535310780)

[5.2.3结束界面函数的介绍 13](#_Toc535310781)

[6调试与测试 14](#_Toc535310782)

[7设计总结 15](#_Toc535310783)

[7.1尚未解决的问题： 15](#_Toc535310784)

[7.2体会： 15](#_Toc535310785)

# 1 设计目的

## 1.1课程设计目的

本课程设计是本专业的一门重要实践性教学环节。在学习了专业基础课和《Python程序设计》课程的基础上，本课程设计旨在加深对Python程序设计的认识，对Python语言及其语言生态有一个进一步的掌握和应用，学会运用Python标准库及外接相关库来解决实际问题的基本能力，培养和提高学生分析问题、解决问题的能力，尤其是提高学生使用Python为开发语言来进行问题描述、交流与思考的能力，为毕业设计和以后的工程实践打下良好的基础。

## 1.2 题目选择：

贪吃蛇小游戏

## 1.3 设计来源

贪吃蛇小游戏的来源是参考经典手机游戏贪吃蛇修改而成，是我们童年的回忆，主要利用玩家的反应来控制蛇身的上下左右移动，来使蛇身逐渐变长，记录得分，挑战自我。由于传统游戏过于简单，界面单一，没有挑战性，我们在传统游戏的基础上进行了很多修改和功能添加。

# 2 课程设计和要求：

## 2.1 课程设计教学任务和要求

本课程设计主要任务是以Python为开发语言完成一个100~300行左右规模的程序项目开发。

课程设计的基本要求是：、设计要求、总体设计，在后期完成详细设计、调试与测试、设计总结等。要求能完成所选题目的主要功能，程序运行健壮、正确，代码要有详细注释，可读性好；所写文档结构合理、内容完整、叙述清晰。更高要求是：有创意、系统界面美观在课程设计的各个阶段严格、规范地完成相关的文档，例如在初期按时完成设计目的。

由于课程设计项目具有一定的综合性，鼓励具有不同特长和不同能力的学生互相组队。项目小组自己推荐一名组长，实行“组长负责制”。组长组织组员进行项目选题、任务分配、方案确定、方案设计、系统调试测试，组员分工协作。小组成员开展项目讨论，互相支持，形成协作意识。

## 2.2课程设计具体要求

1.界面美观，不会产生审美疲劳，不显得大众化。

2.游戏运行流畅，帧数稳定。

3.游戏机制进行创新，游戏玩法多元化。

4.游戏可移植性良好

5.游戏出现的bug不能太多

# 3总体设计

## 3.1 小组人员分配

组员：李秀琛，闵昌茁，黄帅哲

题目：贪吃蛇小游戏 Game of snake

小组位置：github

内容：运用pygame中的方法直接进行界面设计，可以插入图片显示文字，并且添加了控制条件使游戏可以反复运行。实现了传统的贪吃蛇，并且在传统贪吃蛇的玩法上加大了难度。在代码中定义了多个函数，成员可以有明确的分工，最后只需要调用对应的函数即可。

## 3.2 任务分配：

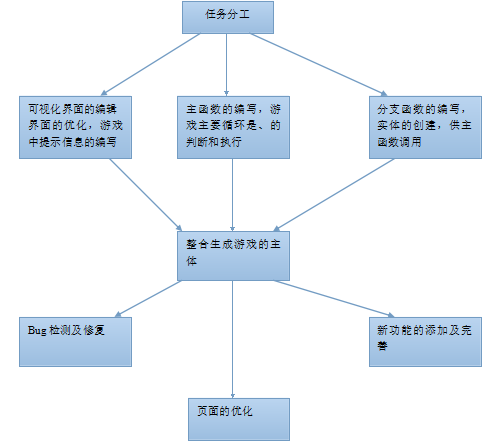
李秀琛：分支函数的变形，总体流程分析，游戏实体的创建

闵昌茁：游戏运行的编写，主函数的设计编写

黄帅哲：游戏开始界面的编写，游戏结束界面的编写，提示信息

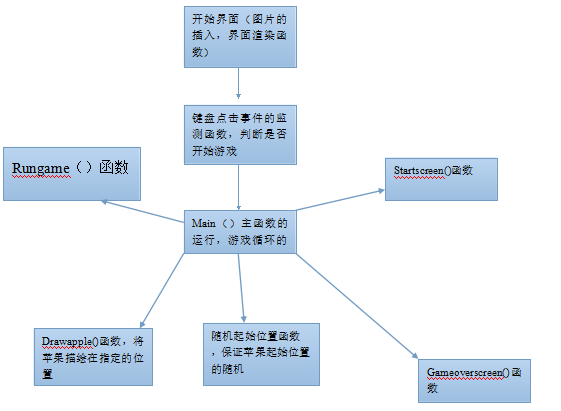
3.3具体功能设计：  
1. 贪吃蛇吃到方块之后蛇身长度变长  
2.贪吃蛇通过“left,right,up ,down”或“上，下，左，右”控制蛇的移动  
3. 在游戏中添加障碍物，蛇碰到后游戏结束  
4. 蛇在碰到边缘时，会从另一边出现，不会结束游戏  
5. 苹果的颜色会根据刷新频率变换颜色，呈现闪烁状  
6. 蛇身颜色按频率刷新  
7. 吃到苹果之后速度加快

**任务分工结构图：**

****

图表 1任务分工结构图

**功能结构图：**



图表 2 功能结构图

# 4 设计实现

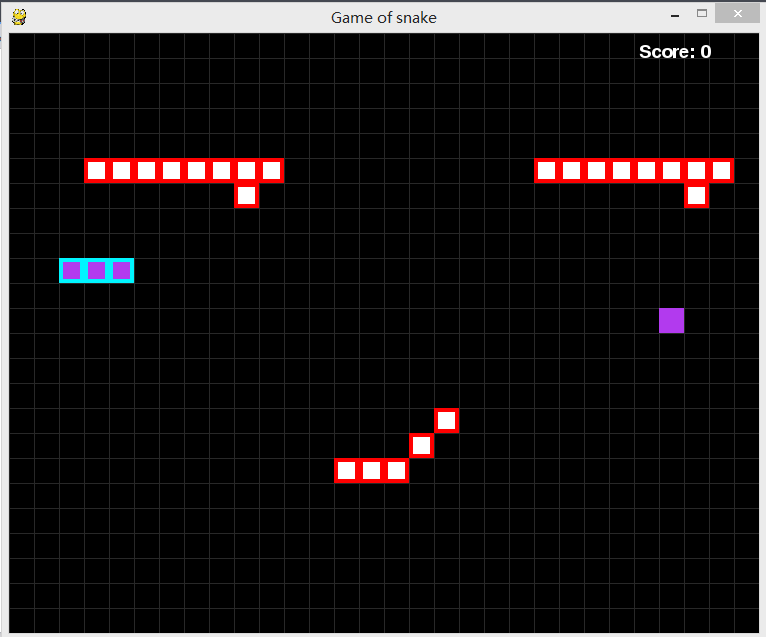
## 4.1 程序运行效果

****

图表 3 开始界面

## 4.2 实现方法

游戏是基于第三方库pygame的可视化界面实现的，使用到了库中的很多函数。

****

图表 4 游戏界面

### 4.2.1 开始和结束界面的编辑

做为一个游戏，肯定是需要加载大量图片的，pygame自带了图片加载功能 ，然后用screen的blit方法把图片放在一个位置。

### 4.2.2 键盘控制的实现

游戏是交互性的，所以我们要捕捉用户的输入，然后给予反馈。这里我们要实现图片的向右移动。老玩家都是会用方向键的，在按下方向键后，就需要更新屏幕，然后把图片更新放在新的坐标，也就是把增加x轴的坐标值。这就是游戏的核心编程思路了，玩家的输入都是要转换成数学公式后更新屏幕。在此基础上进行修改，fill函数就是用RGB颜色更新屏幕。所谓RGB就是三原色的二进制表示， (0,0,0)就是黑色。x就是坐标值，每次按下方向键，我们就递增10。然后显示图片和更新屏幕。这就是一个简陋的游戏了。

## 4.3实现结果评价

优秀：

在最初只预期完成结果只是简单的实现贪吃蛇变长，穿墙而过等一些基本功能。但是在设计的过程中，想法来源不断，于是在不停的实现与调试之后，完成了远远超过预期的功能。虽然还有一些bug没有修复，但是由于此类bug触发条件十分困难，所以对程序的正常运行干预非常小，因此自我评价为优秀。

# 5详细设计

## 5.1界面代码

def drawPressKeyMsg():

pressKeySurf = BASICFONT.render('Press a key to play.', True, darkpurple)

pressKeyRect = pressKeySurf.get\_rect() #获得对象

pressKeyRect.topleft = (WINDOWWIDTH - 200, WINDOWHEIGHT - 30) # 提示信息位置

DISPLAYSURF.blit(pressKeySurf, pressKeyRect)

def showStartScreen():

pic = pygame.image.load('ts.jpg') #添加图片

titleFont = pygame.font.Font('freesansbold.ttf', 50) #设置字体

titleSurf = titleFont.render('Game of snake',True, BLACK)

while True:

DISPLAYSURF.fill(WHITE) #设置背景颜色

DISPLAYSURF.blit(pic, (125,100)) #显示图片

DISPLAYSURF.blit(titleSurf,(180,50)) #显示文字

drawPressKeyMsg() #按任意键后开始

if checkForKeyPress(): #检查按钮事件

pygame.event.get() #清空事件队列

return

pygame.display.update() #画面更新

def showGameOverScreen():

pic = pygame.image.load('gameover.png')

DISPLAYSURF.blit(pic, (0,0))

drawPressKeyMsg()

pygame.display.update()

pygame.time.wait(500) #设置延时，避免结束后直接开始

checkForKeyPress() # 重置按键事件队列

while True: #无限循环按键开始事件

if checkForKeyPress():

pygame.event.get() # 遍历事件，按顺序依次进行不同的操作

return

## 5.2详细介绍

在编写界面前，需要先下载pygame库，在界面设计的部分，大多数使用的都是pygame中的方法。

### 5.2.1提示信息函数介绍

（1）pressKeySurf = BASICFONT.render('Press a key to play.', True, darkpurple)

该行代码中BASICFONT为主函数中定义的全局变量，意义是设置了freesansbold的字体，是系统自带的字体，所以不用下载，可以直接引用。在pygame中直接使用.render方法写字，其中第一个参数是需要写的字，第二个参数是布尔类型的值，表示是否开启抗锯齿，抗锯齿主要是关于文字圆滑度的，当为False关闭抗锯齿时，文字边缘会有毛边，第三个参数是字体的颜色。

（2）pressKeyRect = pressKeySurf.get\_rect()

该行代码使用了.get\_rect()方法，该方法的作用是获取已设定矩形的大小，然后可以再使用.rect方法获取矩形的位置 。

（3）pressKeyRect.topleft = (WINDOWWIDTH - 200, WINDOWHEIGHT - 30)

该行代码确定了矩形的位置，也就是前面输入的字符的位置，即左上角的位置坐标为(WINDOWWIDTH - 200, WINDOWHEIGHT - 30)，这样可以更加方便的改变这一矩形块的位置。

（4）DISPLAYSURF.blit(pressKeySurf, pressKeyRect)

该行代码使用.blit方法，作用是将字显示到界面上，第一个参数是想要显示的信息，第二个参数是左上角的位置。

这一部分是提示信息‘press a key to play’在开始结束界面中的显示，只需要在具体的函数中直接调用即可显示。

### 5.2.2开始界面函数介绍

（1）pic = pygame.image.load('ts.jpg')

该行代码使用了pygame中.image.load方法，作用是添加图片，参数中直接写了图片的名称及后缀名，但是这样写需要将图片放置在同一文件夹内，才可以使用相对路径引用图片。

（2）titleFont = pygame.font.Font('freesansbold.ttf', 50)

该行代码使用了pygame中font.Font方法，用来定义字体以及字体大小，第一个参数为字体名，第二个参数为字体大小。字体可以在网上下载，然后在代码中便可以使用。

（3）titleSurf = titleFont.render('Game of snake',True, BLACK)

该行代码使用了.render方法写字，将Game of snake写在界面上，调整为抗锯齿，字体颜色为黑色。

（4）DISPLAYSURF.fill(WHITE)

该行代码使用.fill方法，填充了背景颜色，参数中的颜色WHITE是在前面定义的RGB颜色的值，否则程序将无法运行。

（5）DISPLAYSURF.blit(pic, (125,100))

该行代码使用.blit方法显示图片，并且在第二个参数设定图片位置。

（6）DISPLAYSURF.blit(titleSurf,(180,50))

该行代码使用.blit方法显示文字，并且设定文字位置。

（7）drawPressKeyMsg()

该行调用了提示信息的函数，直接使提示信息的界面显示在开始界面中，使用起来更加方便，而且要修改提示信息时只需要在对应函数中修改即可完成，使得函数间有较低的耦合度，并且使用的时候也很简单。

（8）while True:

if checkForKeyPress():

pygame.event.get()

return

pygame.display.update()

这段代码的作用是按键后可以进入游戏的界面，在这一段中调用了checkForKeyPress()函数，这个函数主要是检查按键点击事件的。这个循环判断语句是遍历整个整个键盘按键事件，当有按键输入时，会返回值来进行界面的切换。Pygame.display.update()作用是进行页面刷新的，是页面变化的一个条件。

### 5.2.3结束界面函数的介绍

（1）pic = pygame.image.load('gameover.png')

该行代码使用image.load方法插入图片

（2）DISPLAYSURF.blit(pic, (0,0))

该行代码使用.blit方法显示图片，设定图片位置。

（3）drawPressKeyMsg()

该行代码调用函数，使提示信息在结束界面显示。

（4）pygame.display.update()

该行代码刷新了页面，使得页面可以进行变化

（5）checkForKeyPress()

调用了按键检测函数，重置按键事件的队列，这样使得结束界面回到和开始界面一样的条件，可以重新开始游戏。

（6）while True:

if checkForKeyPress():

pygame.event.get()

这段代码同样是键盘按键后开始游戏的判定条件

（7）pygame.time.wait(500)

该行代码使用time.wait方法来设置操作时间延时，在游戏判定结束的条件后，由于有些时候会没有反应过来而继续进行键盘的操作，这个时候由于游戏界面的结束和游戏结束界面的转换时间较短，进行操作之后容易直接跳过结束界面，进行按键事件的判断直接进入下一局游戏，所以设置了操作时间延时。在判定游戏结束后有500ms的时间里，键盘的操作是无效的。

# 6调试与测试

在刚开始的时候，我只是编写了图片和文字的显示方法，在运行的时候可以使图片和文字显示出来，但是在整个代码整合之后，由于需要三个界面之间的切换，随后又增加了按键检测时间。同时也在主函数中设置了判断条件，在开始界面正常运行时，程序才能正常运行。

刚开始使用了pygame中的rotate方法创建了一个矩形块，里面的文字可以随着帧数旋转，但是最终认为不是很美观，又采用了文字与图片结合的方式来绘制开始结束界面。起初由于blit函数的使用顺序不同，导致文字和图片有的被覆盖掉了，才懂得了内容显示的优先级，写在前面的按照执行是最先执行的语句，所以会直接显示在最下方，在后面的代码将会一一覆盖在上方，所以需要位置的协调，获证将有些文字显示在图片上方，直接将文字的blit方法写在图片的方法下解决了这一问题。

在插入图片时，有的图片过大所以只能在界面中显示局部，没有显示完整，在网上搜索的在python中自适应的方法操作复杂，于是直接在本地的画图软件中将图片的分辨率改成了需要的大小，并且将长宽比裁剪成了需要的，但是缺点是在修改完图片之后无法恢复，需要在一开始就确定页面的大小和布局，否则在之后的修改中将会十分复杂。并且在添加图片的过程中，一开始使用相对路径的时候总是格式不对，后来查询之后发现python中添加图片的相对路径，只需要在一个文件夹中，然后参数中仅仅只需填入图片名.后缀名即可。

在代码整合之后，游戏会出现结束之后直接重新开始的bug，不显示结束界面，经过分析之后，发现是因为输入方式是遍历的方式产生的，当键盘输入时，计算机会记录按键队列，然后按顺序执行，当判定游戏结束时有可能队列中会残留按键信息，所以进入结束界面时按键队列直接输入，导致直接从结束界面进入了重新开始的游戏界面。在网上查找资料后，使用了time.wait方法，使得进入结束页面后判定重新开始游戏会延时，这样游戏结束后就不会直接开始了。

# 7设计总结

## 7.1尚未解决的问题：

1. 图片无法随着窗口的大小而改变
2. 窗口无法改变大小

## 7.2体会：

在本次课程设计中，通过小组合作，不仅学到了python，还学到了很多人生智慧。仅靠我一个人的能力要做出这样的程序，是需要很长时间的，而我们一个小组，只要分工妥当，效率比一个人要高了非常多。在编写代码的过程中，不仅可以学到自己分工内的知识，而且在同伴有疑问时，共同解决问题的过程也是成长，比如在帮助同伴检测修复bug时，也锻炼了自己的逻辑，同时对相应的解决方案也有了一定的了解，锻炼了自己的计算思维。

在编写代码时也发现了python语言的简便性，很多库中的方法都是封装好的，可以直接在命令行中下载相应的库，然后通过import调用库就可以在后续直接使用了。同时很多代码的含义与英语的意思也都很相近，可读性非常好。在网上搜索资料后就可以很轻易的绘制出来一个界面，但是比较难的是如何进行各个部分的连接，需要有一些逻辑的先后顺序在里面，否则进程就会出现问题。

在学习中，我觉得最重要的就是会学习。首先就是要学会问问题，有些思想自己半天想不出的观点，别人可能一下子就可以想到，所以请教别人是一个很好的学习方式。但是在请教别人之后，不能直接照搬照抄写上去就不管了，需要把别人的知识消化吸收变成自己的知识，这样才会有长进。如果是自己独自学习，首先要了解的就是机制，只有懂得了语言的机制，这样套路才能更顺利的运用出来，许多语句都是要求我们用不同于数学的计算思维，这与我们之前通俗的思考问题方式不同，我们需要先转变自己的思考方式，才可以更好的深入学习python这门语言。

同时在课堂上，老师更注重我们的动手能力，给我们很多机会来锻炼编程，从一开始的Hello World到后来不断复杂的代码，都是我们一次次的挑战，通过不断的练习计算思维，让我在设计程序中能够很好的理清思路，在运用pygame库时也能很轻易的进行调用，对方法的实用。

这次完成了贪吃蛇这个课程设计，最重要的就是进行了小组的配合，在之前的学习中一直都是自己独自完成的工作，这一次的课设第一次采取了小组的方式。所以我们才有机会不断的去思考不断的去实现新的功能，正是因为组合的力量更强大，我们才能在这么短的时间内尽力去完成、去调试这个程序。单凭一个人的力量，总会要付出更多的时间来完成，让我明白了团队的重要性。