**XXXXXX·XXXXXX学院**

**《Python程序开发课程设计》报告**

贪吃蛇游戏的设计与实现

专 业： 软件工程

班 级： XXXXXXXXXXX

学 号： XXXXXXXX

姓 名： rv

指导老师： XX

报告日期： 2022年06月

目录

[一、 概述 3](#_Toc27245)

[二、 系统分析 3](#_Toc24701)

[三、 概要设计 3](#_Toc23109)

[1． 设计要求 3](#_Toc21298)

[2． 设计思想 4](#_Toc1803)

[3． 整体框架 4](#_Toc23380)

[4． 流程图 4](#_Toc27633)

[四、 详细设计 5](#_Toc15071)

[（一） 设计思想 5](#_Toc4146)

[（二） 具体模块实现 6](#_Toc15508)

[五、 调试过程及运行结果 11](#_Toc6738)

[1. 调试过程 11](#_Toc24730)

[2. 实验结果 11](#_Toc5801)

[六、 结束语 12](#_Toc9079)

[七、 参考资料 12](#_Toc21256)

[八、 致谢 13](#_Toc179)

1. **概述**

课程设计是实践性教学中的一个重要环节， 它以某一课程为基础， 可以涉及和课程相关的各个方面， 是一门独立于课程之外的特殊课程。 课程设计是让同学们对所学的课程更全面的学习和应用， 理解和掌握课程的相关知识。 本次课程设计是以学生独立思考解决问题为主， 教师指导为辅， 结合上机操作， 完成指定的任务， 做出设计报告。

Python是一门重要的专业基础课， Python课程设计是教学实践环节中一项重要内容， 进行此课程设计旨在掌握基础知识的基础上， 进一步加深对 Python的理解和掌握； 提高和加强学生的计算机应用及软件开发能力， 使学生具备初级程序员的基本素质； 培养学生独立分析问题、 解决问题、 查阅资料以及自学能力， 以适应信息管理行业日新月异的发展形势。

本次课设我选择了贪吃蛇游戏的设计与实现。

贪吃蛇游戏要求一条蛇在密闭的空间内， 在空间内随机出现一个食物， 蛇向上下左右四个方向移动， 蛇头撞到食物， 则表示食物被吃掉， 这时蛇的身体长一节； 接着又出现食物， 等待被蛇吃掉。 同时， 场中还有随机炸弹。 如果蛇在移动过程中， 撞到墙壁、 炸弹或身体交叉（蛇头撞到自己的身体）， 则游戏结束。

1. **系统分析**

贪吃蛇程序关键在于表示蛇的图形及蛇的移动。 用一个小矩形快表示蛇的一节身体， 身体每长一节， 增加一个矩形块。 移动时必须从蛇头开始， 蛇不可以向相反的方向移动， 如果不按任意键， 蛇自行在当前方向前移， 但按下有效方向键后， 蛇头朝该方向移动， 一步移动一节身体， 所以按下有效方向键后， 先确定蛇头的位置， 而后蛇的身体随蛇头移动。

图形的实现是从蛇头新位置开始画出蛇， 这时， 由于未清屏的原因， 原来的蛇的位置和新的位置差一个单位， 看起来蛇多一节身体， 所以再将蛇的最后一节去掉， 来实现蛇的移动。 食物的出现与消失也是画矩形块和清除矩形块。

为便于后续操作， 在设置蛇、 食物和炸弹时， 让每一个点都是由行和列组成。 用列表存储蛇的身体， 便于蛇身体内小方块的增加和删除。 然后利用图形驱动， 制作美观的游戏界面， 通过随机函数产生随机的食物, 控制游戏过程食物的出现。 定义键盘操作控制游戏过程蛇的移动方向， 画出边界, 并判断游戏是否结束。

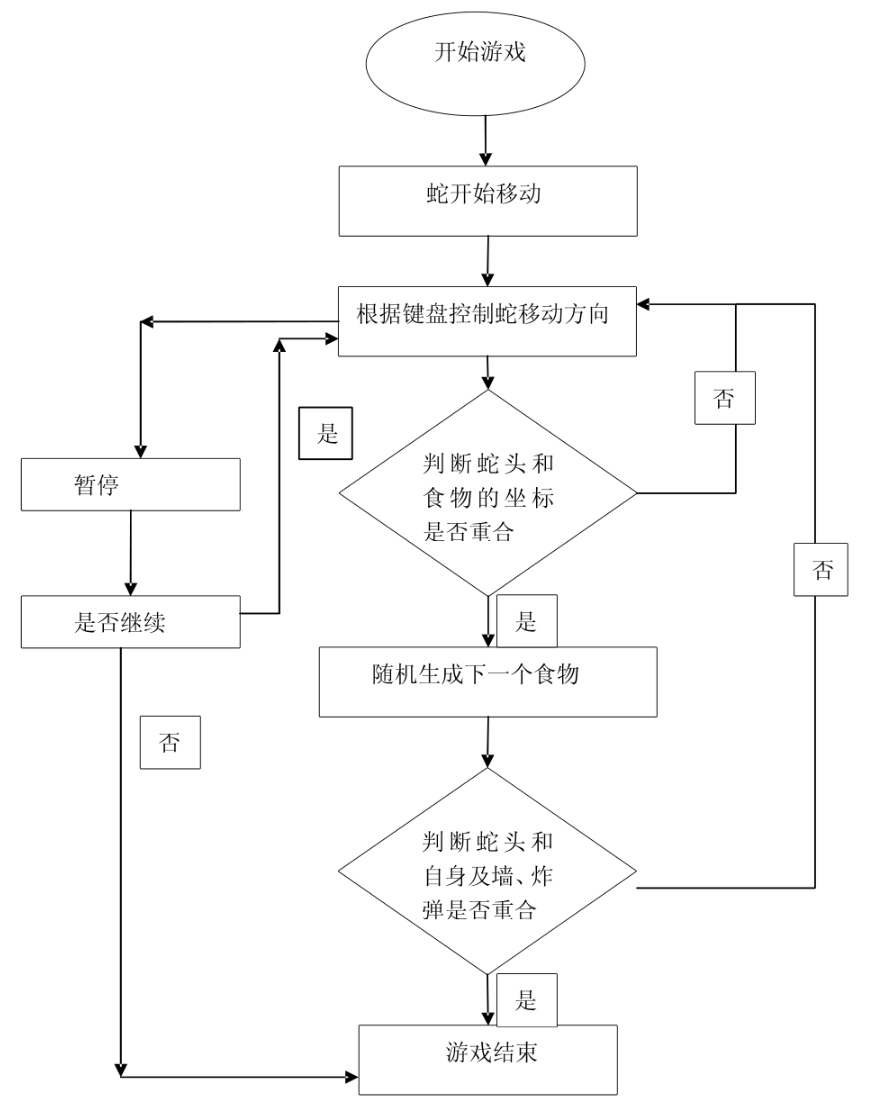
1. **概要设计**
2. **设计要求**

编写程序实现贪吃蛇游戏， 贪吃蛇游戏是一个深受人们喜欢的经典游戏： 具体要求有以下几点：

1. 对系统进行功能模块分析、 控制模块分析正确， 符合课题要求， 实现相应功能； 可以加以其他功能或修饰， 使程序更加完善、 合理；
2. 系统设计要实用， 采用模块化程序设计方法， 编程简练、 可用， 功能全面；
3. 说明书、 流程图要清楚；
4. **设计思想**
5. 可以暂停/继续游戏（通过特定按钮实现）
6. 通过键盘上的四个光标键控制蛇的移动
7. 能够区分蛇头蛇尾
8. 蛇的身体能够随着吃到食物的数量逐渐变长
9. 游戏中设有随机炸弹，蛇碰到后死亡
10. 要求在游戏的过程中添加背景音乐和发生特殊事件的特效音（吃到食物、碰壁等）。
11. **整体框架**

本程序主要包含5个主要模块：

1. 创造模块 —— 实现游戏窗口的创建， 以及蛇、 食物及炸弹的创造。
2. 进食模块 —— 实现蛇吃到食物后的各种变化。 如身体的变化、 食物的变化及炸弹的变化。
3. 死亡模块 —— 判断蛇是否死亡。 当贪吃蛇碰到墙壁、 碰到炸弹， 或碰到自己的身体时，游戏结束。
4. 音乐模块 —— 实现游戏进行时、 吃到食物时以及蛇死亡时的音乐播放。
5. 主函数模块 —— 进行各模块之间的调用， 且实现人机交互。 通过键盘操作控制游戏的开始、 暂停、 结束， 以及游戏过程中过程蛇的移动方向。
6. **流程图**



1. **详细设计**

（一） **设计思想**

首先我们都知道游戏最基本的功能有开始， 暂停， 继续， 退出， 所以首先我用了Panel Event事件模块来使游戏能用键盘敲击触发特定的情节。 用Pygame 的 display模块设置游戏的界面运动空间， 即游戏窗口， 来实现贪吃蛇的运动空间。

要定义类和函数， 来实现蛇运动时的一些属性与方法， 首先明白这个游戏实现的是贪吃蛇吃食物， 所以还需要定义食物， 食物被吃后要实现食物的重建。 玩游戏时还需要有控制键的设置， 这一模块可以选用循环语句来实现。

实现基本的功能后要注意怎样的情况会造成蛇的死亡， 可以想到的有越界线， 咬到自己都会死亡。 随着思想的成熟， 觉得只吃食物好像太简单， 所以在这个贪吃蛇游戏的基础上又设置了随机的炸弹， 这样游戏就更有意思了。

（二） **具体模块实现**

**1、 Point类定义**

1. class Point:   *#希望每一个点都是由行和列组成*
2. row=0
3. col=0
4. def \_\_init\_\_(self,row,col):
5. self.row=row
6. self.col=col
7. def copy(self):    *#蛇身操作时要用*
8. return Point(row=self.row,col=self.col)

**2、 创造模块**

（1） 创造蛇模块

蛇头

1. head = Point(row=int(ROW/2), col=int(COL/2))  *#蛇头*
2. head\_color = (0, 128, 128)  *#浅蓝色*

蛇身

1. snakes = [
2. Point(row=head.row, col=head.col+1),
3. Point(row=head.row, col=head.col+2),
4. Point(row=head.row, col=head.col+3),
5. ]
6. snakes\_color = (200, 200, 200)  *#灰色*

（2） 创造食物模块

通过坐标点的方法来实现食物， 食物是随机出现的， 而且一旦被吃掉就会立刻创建新的食物。

1. def creat\_point():
2. '''
3. randint 产生的随机数区间包含左右极限
4. 即左右都闭：[x,y]
5. randrange 产生的随机数只包含左极限
6. 即左闭右开：[x,y)
7. randint 产生的随机数是在指定的某一个区间内的一个值
8. randrange 产生的随机数可以设定一个步长，也就是一个间隔。
9. '''
10. point = Point(0, 0)
11. flag = True
12. while flag:
13. point\_row = random.randint(0, ROW-1)
14. point\_col = random.randint(0, COL-1)
15. point = Point(point\_row, point\_col)
16. *#如果点随机产生在蛇身上，重新创造点*
17. flag = False
18. if point.row == head.row and point.col == head.col:
19. flag = True
20. for snake in snakes:
21. if point.row == snake.row and point.col == snake.col:
22. flag = True
23. break
24. return point

（3） 创造炸弹模块

创建炸弹和创建随机食物思路是一致的， 都是在一定范围内随机出现。一旦被食物吃掉就会立刻创建新的炸弹。且需要判断炸弹是否与食物重合。

1. def creat\_bomb(food):
2. flag = True
3. while flag:
4. bomb = creat\_point()
5. if bomb.col != food.col or bomb.row != food.row:
6. flag = False
7. return bomb

**3、 进食模块**

（1） 如果吃到食物，则生成新的食物和炸弹，接着处理蛇身

1. *#吃东西*
2. eat = head.row == food.row and head.col == food.col
3. if eat:
4. food = creat\_point()
5. bomb = creat\_bomb(food)
6. eat\_music()
7. #处理蛇身
8. body(eat)

（2） 蛇身的处理

1. def body(eat):
2. '''
3. 处理身子
4. 1.把原来的头插入到snakes的头上
5. 2.如果未吃到食物，把snakes的最后一个删除
6. '''
7. snakes.insert(0, head.copy())
8. if not eat:
9. snakes.pop()

**4、 死亡模块**

判断蛇是否死亡。当贪吃蛇碰到墙壁、碰到炸弹或碰到自己的身体时，游戏结束。

1. def check():
2. '''
3. 检测
4. 1.撞墙
5. 2.撞炸弹
6. 3.撞自己
7. '''
8. dead = False
9. if head.col < 0 or head.col >= COL or head.row < 0 or head.row >= ROW:
10. dead = True
11. elif head.col == bomb.col and head.row == bomb.row:
12. dead = True
13. else:
14. for snake in snakes:
15. if head.col == snake.col and head.row == snake.row:
16. dead = True
17. break
18. return dead

**5、 游戏结束模块**

当游戏结束时，在控制台打印出“游戏结束”，且播放结束音乐并退出程序。

1. def is\_dead():
2. print("游戏结束")
3. dead\_music()
4. pygame.quit()
5. sys.exit(0)

**6、 音乐模块**

实现游戏进行时、 吃到食物时以及蛇死亡时的音乐播放。

1. def start\_music():
2. *# 加载背景音乐并播放*
3. file = r"C:\Users\Rose\_Vanderboom\Desktop\贪吃蛇-rv\开始.mp3"
4. pygame.init()  *# 进行全部模块的初始化，*
5. pygame.mixer.init()  *# 只初始化音频部分*
6. pygame.mixer.music.load(file)  *# 使用文件名作为参数载入音乐*
7. pygame.mixer.music.set\_volume(1)
8. pygame.mixer.music.play(-1)  *# 播放载入的音乐。该函数立即返回，音乐播放在后台进行。   -1:循环播放*
9. def eat\_music():
10. file = r"C:\Users\Rose\_Vanderboom\Desktop\贪吃蛇-rv\eat.wav"
11. eat\_sound = pygame.mixer.Sound(file)
12. eat\_sound.set\_volume(2)
13. eat\_sound.play()
14. def dead\_music():
15. *# 加载背景音乐并播放*
16. file = r"C:\Users\Rose\_Vanderboom\Desktop\贪吃蛇-rv\Toby Fox - Shop.mp3"
17. pygame.mixer.init()  *# 只初始化音频部分*
18. pygame.mixer.music.load(file)  *# 使用文件名作为参数载入音乐*
19. pygame.mixer.music.set\_volume(2)
20. pygame.mixer.music.play(-1)  *# 播放载入的音乐。该函数立即返回，音乐播放在后台进行。   -1:循环播放*
21. time.sleep(6)

**7、 主函数模块**

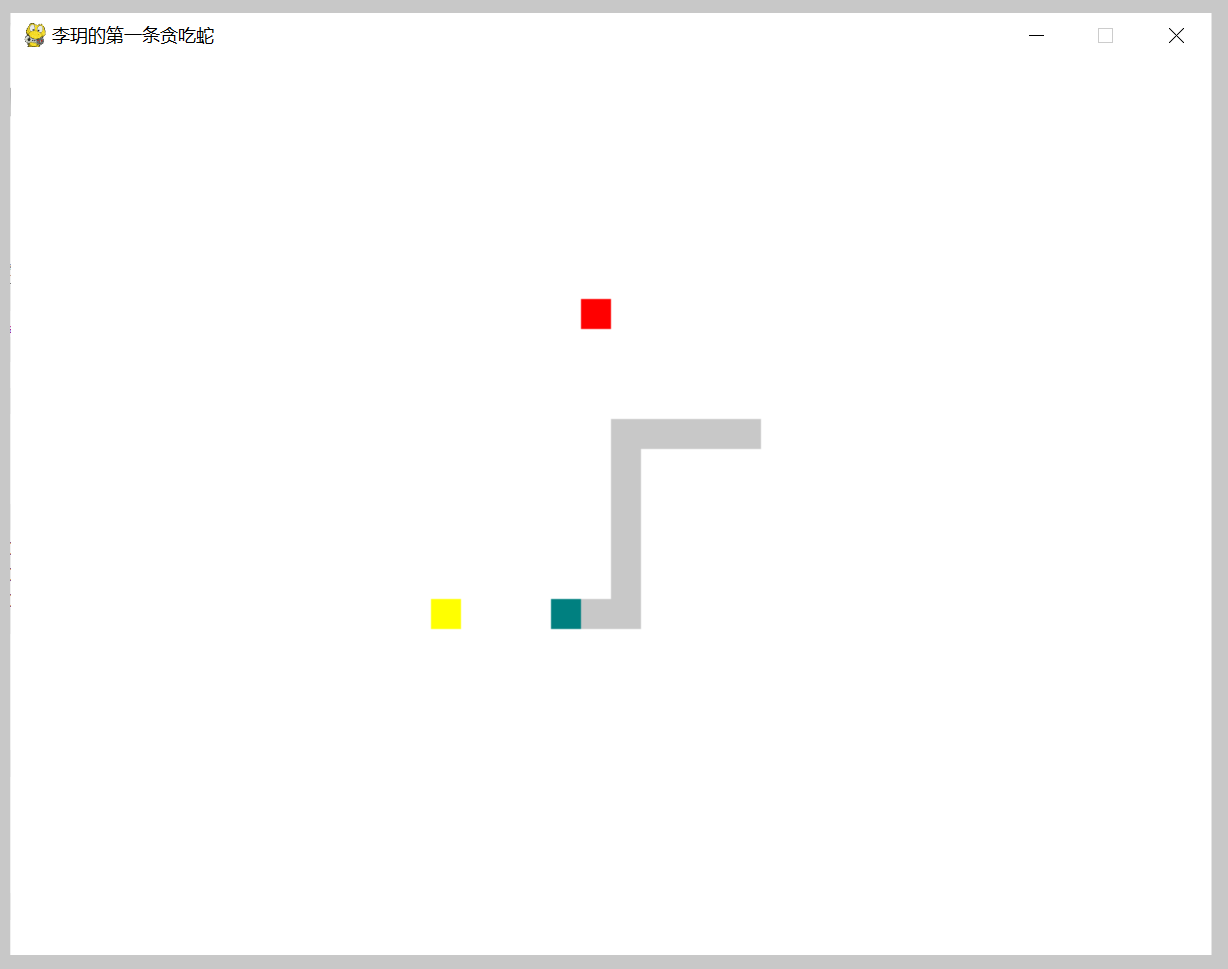
1. def main():
2. global food, bomb
3. *#游戏循环*
4. *#启动游戏时的相关变量初始化*
5. start\_music()   *#放音乐*
6. direct = 'left'
7. game\_start = False  *#当start = True，over = True 时 -- GAME OVER*
8. game\_over = True
9. game\_pause = False
10. clock = pygame.time.Clock()
11. draw()  *#画出来*
12. while True:
13. for event in pygame.event.get():  *# 获取当前发生的事件（队列）*
14. *#print(event)*
15. if event.type == pygame.QUIT:  *# 退出*
16. pygame.quit()
17. sys.exit(0)
18. elif event.type == pygame.KEYDOWN:    *#键盘*
19. if event.key == 13:   *#回车键*
20. if game\_over is True:
21. game\_start = True
22. game\_over = False
23. elif event.key == 32:
24. if not game\_over:
25. game\_pause = not game\_pause
26. elif game\_start is True and game\_over is False and game\_pause is False:
27. if event.key == 1073741906 or event.key == 119:   *# wasd 和 上下左右 都可操作*
28. if direct == 'left' or direct == 'right':
29. direct = 'up'
30. elif event.key == 1073741905 or event.key == 115:
31. if direct == 'left' or direct == 'right':
32. direct = 'down'
33. elif event.key == 1073741904 or event.key == 97:
34. if direct == 'up' or direct == 'down':
35. direct = 'left'
36. elif event.key == 1073741903 or event.key == 100:
37. if direct == 'up' or direct == 'down':
38. direct = 'right'
39. if game\_start is True and game\_over is False and game\_pause is False:
40. *#吃东西*
41. eat = head.row == food.row and head.col == food.col
42. if eat:
43. food = creat\_point()
44. bomb = creat\_bomb(food)
45. eat\_music()
46. *#处理蛇身*
47. body(eat)
48. *#移动/处理蛇头*
49. move(direct)
50. *#判断蛇是否死亡*
51. dead = check()
52. if dead:
53. is\_dead()
54. draw()  *#画出来*
55. pygame.display.flip()  *#让出控制系统*
56. clock.tick(15)  *# 设置帧频(锁帧)  (1000ms/15)sleep*
57. **调试过程及运行结果**

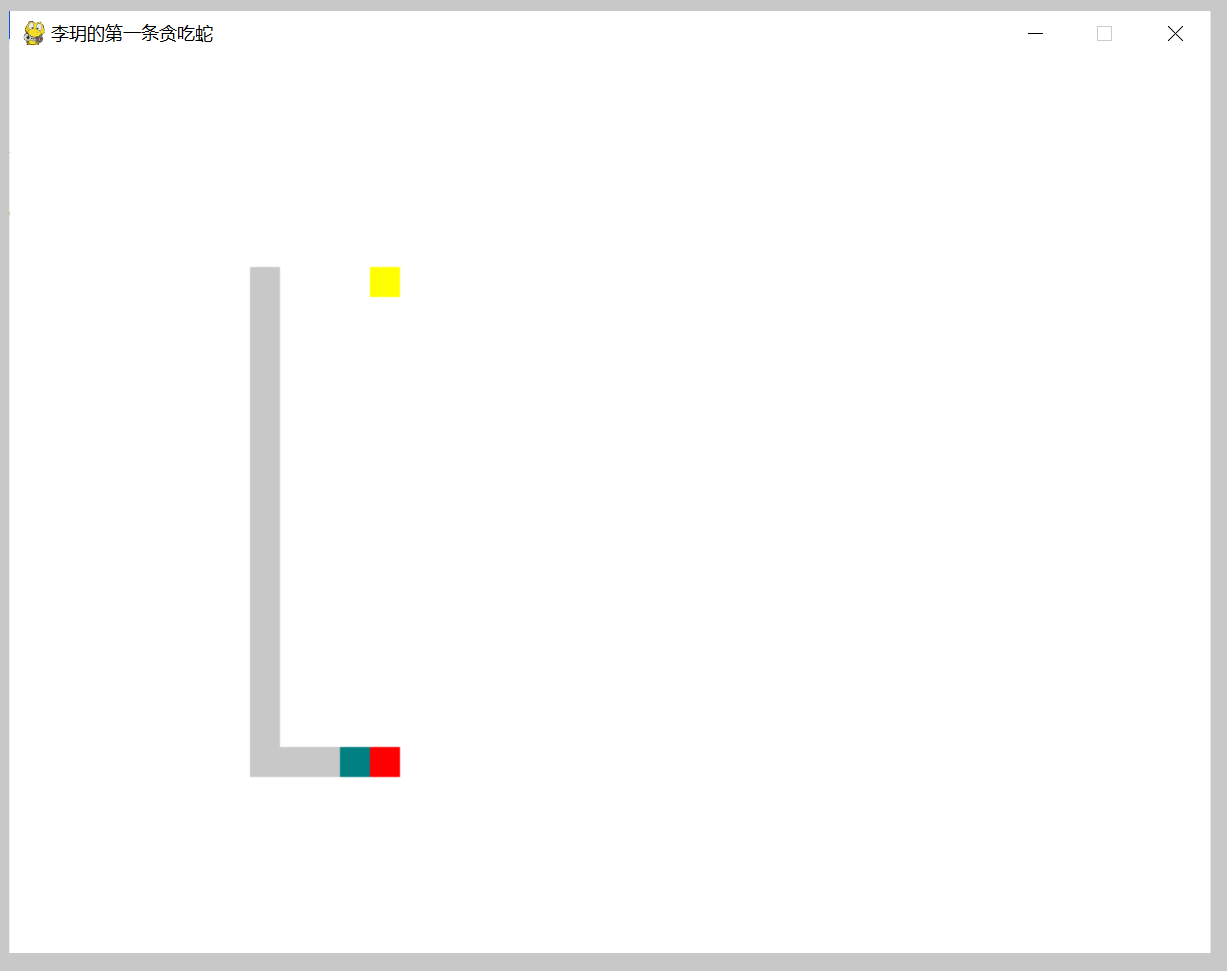
**1. 调试过程**

（1） 在实现撞障碍物的过程中， 刚刚开始的时候墙不能起作用， 贪吃蛇有时能穿过障碍物， 后来才发现时障碍物的设置范围出现了错误。 经过不停的尝试与计算才设置出正确的范围。

（2） 在玩游戏的过程中发现有炸弹与食物重合的情况， 通过分析原因， 这与构造食物时判断食物与是否与蛇重合的道理是一样的， 这里需要加一个循环语句来判断， 如果炸弹创造在食物上的话，则重新创造炸弹，直到炸弹不与食物重合。

**2. 实验结果**

游戏运行主界面



1. **结束语**

这次的课设对我来说有很大的收获， 我学到了许多课本上没有学到的知识， 这次我将将课本上的理论知识充分的转化成了 一些实用技能。 通过这次自己制作的游戏， 丰富了自己的实践技能， 扩展课本专业的知识面。 在这次设计的同时， 我又从中学到了 许多东西。 但由于我对这样的开发还只是一个开始， 了解的不多， 这其中或许还有很多的不足， 有些模块做的不是很好， 界面的设计及整体的布局还是比较简单。

此外， 在这一次的课程设计过程中， 我上网查阅了大量的资料， 对 python有了更深的理解，对于软件工程这些辅助性的教材也巩固了不少， 为我这次的课设提供了很大的帮助， 锻炼了我的能力。 更重要的是， 它还让我认识到了自己的不足之处， 在编程方面， 我还是比较欠缺，以后的路任重道远， 我需要的事不断地丰富自己， 充实自己， 这样才能在程序设计方面有所收获。

最后感谢老师在这次课程设计中对我们的悉心教导， 通过这次学习， 让我们掌握了很多有关 Python 的应用知识。

1. **参考资料**

[1] [美]Jonh Zelle. **Python程序设计（第3版）**［M］.王海鹏，译. 北京：人民邮电出版社. 2018.

# [2][美][Jonathan S.Harbour](https://search.kongfz.com/item_result/?status=0&select=2&author=hk4ak6fk6ek61k74k68k61k6ek20k53k2ek48k61k72k62k6fk75k72" \t "https://item.kongfz.com/book/_blank). PYTHON游戏编程入门［M］.李强，译 .北京：人民邮电出版社, 2015.

1. **致谢**

感谢XXXXX一年来对我的辛苦培育， 让我在大学这一年来学到很多东西，特别感谢XXXXX学院为我提供了良好的学习环境、感谢领导、老师们对我无微不至的关怀和指导， 让我得以在这一年中学到很多有用的知识。 在此， 我还要感谢班里同学和朋友， 感谢你们在我遇到问题时的时候帮我解惑， 给予我莫大的支持和鼓励。

特别感谢我本次课程设计的指导老师XXX， 在本系统开发中给予我悉心指导， 从系统开发到结束中过程遇到很多困难都是X老师给我鼓励与指引，使我能够克服重重困难， 将课程设计制作完成。 XXX老师严谨的教学态度、渊博的学术知识、诲人不倦的敬业精神以及宽容的待人风范使我获益颇多。 在此谨向XXX老师致以诚挚的谢意和崇高的敬意。谢谢!

同样， 特别感谢我的任课老师 XXX。 X老师治学严谨，学识渊博，思想深邃，视野雄阔，为我营造了一种良好的精神氛围。 授人以鱼不如授之以渔， 置身其间， 耳濡目染，潜移默化，使我不仅接受了全新的思想观念，还树立了宏伟的学术目标，领会了基本的思考方式。 课本上的知识经由X老师悉心的点拨，再经思考后的领悟，常常让我有“醍醐灌顶”的感觉。

最后，再次感谢与我并肩作战的舍友与同学们，感谢关心我支持我的朋友们，感谢学校领导、老师们，感谢你们给予我的帮助与关怀; 感谢XXXXXX，特别感谢XXXX学院这一年来为我提供的良好学习环境，谢谢!