**《Python程序设计》**

**课程设计报告**

　　　　　　　　　　（2018 —2019 学年第 一 学期）

题 目 兔子与獾游戏设计

学生姓名 丁 萌

专业班级 信管16-2

学生学号 311609030202

教师姓名 徐 文 鹏

成 绩**：**

评 语：

教师签名：

日期：

目录

[一 、设计目的 1](#_Toc535306949)

[1.1课程设计目的 1](#_Toc535306950)

[1.2本课程设计具体目的 2](#_Toc535306951)

[二、设计任务和要求 2](#_Toc535306952)

[2.1课程设计教学任务和要求 2](#_Toc535306953)

[2.2本课程设计具体任务和要求 2](#_Toc535306954)

[三、总体设计 3](#_Toc535306955)

[3.1流程图 3](#_Toc535306956)

[3.2功能图 4](#_Toc535306957)

[3.3．分配情况 5](#_Toc535306958)

[四、设计实现 6](#_Toc535306959)

[4.1最终实现结果 6](#_Toc535306960)

[4.2实验结果评价 7](#_Toc535306961)

[五、详细设计 8](#_Toc535306962)

[5.1输入输出分析 8](#_Toc535306963)

[5.2程序流程图 9](#_Toc535306964)

[5.3算法说明 10](#_Toc535306965)

[六、调试与测试 15](#_Toc535306966)

[6.1调试方法 15](#_Toc535306967)

[6.2测试结果的分析 15](#_Toc535306968)

[6.3问题及解决方法 16](#_Toc535306969)

[七、设计总结 17](#_Toc535306970)

# 一 、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本课程设计具体目的

这个题目是我们主要根据植物大战僵尸这个游戏想出来的。当我们选择题目的时候，想了很久，然后想到小时候经常玩的植物大战僵尸，忽然就有了灵感，所以我们决定根据植物大战僵尸来设计我们的游戏。一是想更好的完成这次课设，加深对游戏设计的理解，更加理解pygame库的使用，算是我们学习的总结，二也是为了回忆儿时玩游戏的感觉，同时了解一下这类游戏的基本功能的实现过程。

# 二、设计任务和要求

## 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：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、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。

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本课程设计具体任务和要求

我们所选题目是兔和獾的2D游戏，具体任务分为设置窗体，添加背景，使兔子能够随着按键移动、实现音效；获取鼠标使得兔子根据鼠标旋转，点击鼠标使兔子射箭，添加被攻击的獾；处理獾与箭头的碰撞、添加健康值和时间的显示、判定赢或输这几部分。

要求是可以实现兔随按键的移动，随鼠标而转动方向，当有獾出现时兔子射箭攻击獾，未射中时生命值会有相应的减少，同时加有计时器，来记录游戏时间，进而可以以此判断玩家的输赢。

# 三、总体设计

## 3.1流程图

玩家是否点击鼠标

否

兔子不射箭

是

兔子射箭

生命值减少

是否射中

否

是

生命值是否为0

獾减少

否 是

游戏结束，玩家失败

游戏继续，计时增加

否

时间是否为90s

是

游戏结束，玩家胜利

## 3.2功能图

决定输赢

时间

加入獾

加入兔子

生命值的改变

碰撞

射箭

鼠标控制转向

键盘控制方向

声效

背景

窗体

兔子与獾

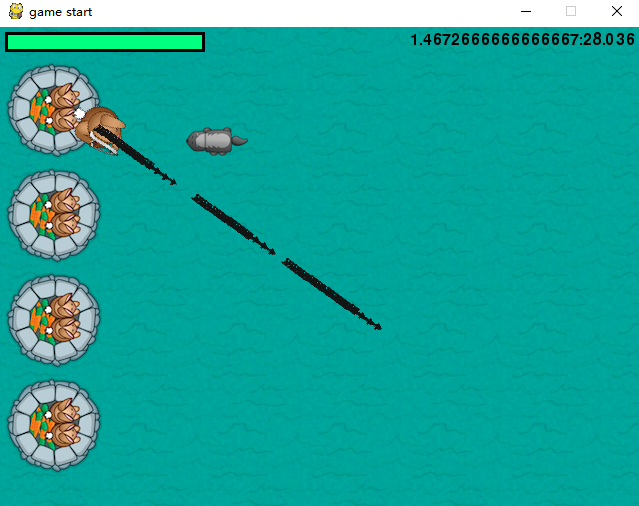
3.3．分配情况  
周紫会：设置窗体，背景，添加堡垒，加入兔子，使兔子能够移动、加入音效；  
刘柯柯：获取鼠标使得兔子根据鼠标旋转，点击鼠标使兔子射箭，添加被攻击的獾；  
丁萌：处理獾与箭头的碰撞、添加健康值和时间的显示、判定赢或输；

# 四、设计实现

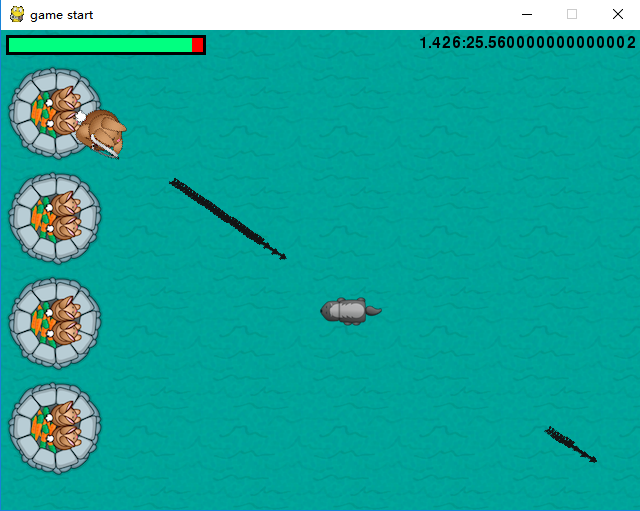
## 4.1最终实现结果



4-1游戏启动界面效果图



4-2主要功能界面-箭的射出效果图



4-3主要功能界面-箭与獾的碰撞效果图



4-4游戏结束界面效果图

## 4.2实验结果评价

最终实现的结果与我们当初小组讨论分析的课题设计要求对比，我们组已经全部实现了当初的设计想法和要求。当初，我们组想的是应当有游戏的背景、音效，设定玩家人物，当有敌人进攻的时候能够反击，并能够决定游戏的输赢，最终实验的结果都已实现了这些功能。

我们小组对最终实现结果的评价是优秀。因为我们小组最终的设计成果可以100%的实现最初的设计想法和要求，比如：箭的射出、箭与獾的碰撞等。总而言之，我们组的对我们实现的成果都很满意，不仅在功能上完成了我们的预想，而且在页面的设计上我们也很满意。

# 五、详细设计

## 5.1输入输出分析

在添加生命条方面，我通过加载生命条的图片，并根据生命条程度添加绿色，最终在页面上呈现出一个绿色的矩形，就是生命条。在实现獾与城堡撞击方面，

输入为獾与城堡撞击，输出为生命值减少，獾消失。在实现獾与箭撞击方面，输入为獾与箭头撞击，输出为獾消失，箭头也同时消失。在游戏结束判定方面，输入为时间或者为生命值，输出为游戏胜利或者游戏失败。

## 5.2程序流程图

獾与箭是否碰撞

獾撞击城堡，城堡生命值减少

否

是

城堡生命值是否为0

箭和獾同时消失

游戏结束，玩家胜利

时间是否达到90s

游戏继续，计时增加

游戏结束，玩家失败

是

否

否

是

## 5.3算法说明

（1）、添加生命值

这一部分主要用了blit()方法，第一个参数为要画的图片，第二个参数为画图片的位置，这里的意思就是在（5,5）位置上画上生命值条的图片，接着根据城堡的生命值往生命条里面添加绿色

#设初值

healthvalue=194

#加载健康图片

healthbar = pygame.image.load("resources/images/healthbar.png")

health = pygame.image.load("resources/images/health.png")

screen.blit(healthbar, (5,5))

for health1 in range(healthvalue):

screen.blit(health, (health1+8,8))

（2）、獾与城堡的撞击

要保证獾冲过来并且在碰到城堡的时候獾就会消失。同时，当獾撞上城堡之后，城堡的生命值就会减少（之后会有设计）。此时，如果獾的x坐标离右边小于64也就是獾撞上了城堡，就删除獾并且减少城堡的生命值，减少的大小为5至20里的一个随机数。（在这里引用了random 库）

在此之前，我查阅了相关的pycharm的一些类：Rect类，表示一个矩形区域的类，用来操作图形、图像等元素，我用get\_rect来获取獾的位置大小信息

badrect=pygame.Rect(badguyimg.get\_rect())

badrect.top=badguy[1]

badrect.left=badguy[0]

if badrect.left<64:#判断是否小于64，即是否撞上城堡

healthvalue -= random.randint(5,20)#生命值减少（5,20）里的一个随机数

badguys.pop(index)#獾的个数相应减少

（3）、獾与箭头的撞击

当獾撞上箭，也就要消失，同时箭也要消失，这里与獾和城堡相撞大致相同，其中，有一点不同是这个if表达式使用了PyGame内建功能来检查两个矩形是否交叉，即判断獾与箭是否相撞，最后一行的blit（）方法是将獾的图像绘制到badguyimg上，最后成一个图像

index1=0

for bullet in arrows:

bullrect=pygame.Rect(arrow.get\_rect())

bullrect.left=bullet[1]

bullrect.top=bullet[2]

if badrect.colliderect(bullrect):#判断獾与箭所在矩形是否交叉，即是否相撞

acc[0]+=1

badguys.pop(index)

arrows.pop(index1)

index1+=1

index+=1

for badguy in badguys:

screen.blit(badguyimg, badguy)

（4）、添加时钟

其中我先设置了字体，然后我用pygame.time.get\_ticks()获取以毫秒为间隔的时间，用render()方法显示，其中，该方法的第一个参数为显示的内容也就是时间，第二个参数设置的True，目的在于让字体比较平滑，第三个参数我设置了字体的颜色为黑色。最后，修改时间条位置，用blit()方法将时间条画在修改后的位置

#设置字体None为pycharm默认的字体样式，24为字号

font = pygame.font.Font(None, 24)

survivedtext=font.render(str((90000-pygame.time.get\_ticks())/60000)+":"+str((90000-pygame.time.get\_ticks())/1000%60).zfill(2), True, (0,0,0))

textRect = survivedtext.get\_rect()

textRect.topright=[635,5]#topright即（right,top）

screen.blit(survivedtext, textRect)

（5）、游戏结束判断

判定输赢的条件，即如果游戏时间超过90s，游戏结束，游戏胜利，如果生命值小于0，游戏结束，游戏失败

第一个if表达式是检查是否时间到了，当游戏时间达到90s，running=0，exitcode=1。第二个是检查城堡是否被摧毁了，当生命值小于0时，running=0，exitcode=0。第三个计算精准度。之后，一个if表达式是检查是赢了还是输了，当exitcode=0时，玩家游戏结束，游戏失败；当exitcode=1时，玩家游戏结束，游戏胜利，然后显示出相应的图片。

pygame.time.get\_ticks()获取以毫秒为间隔的时间，用render()方法显示，其中该方法的第一个参数为显示的内容也就是精准度，第二个参数设置的True，目的在于让字体比较平滑，第三个参数设置了字体的颜色为红色。blit()方法用于将图片放在指定的位置。

running = 1

exitcode = 0

if pygame.time.get\_ticks()>=90000:

running=0

exitcode=1

if healthvalue<=0:

running=0

exitcode=0

if acc[1]!=0:

accuracy=acc[0]\*1.0/acc[1]\*100

else:

accuracy=0

if exitcode==0:

pygame.font.init()

font = pygame.font.Font(None, 24)

text = font.render("Accuracy: "+str(accuracy)+"%", True, (255,0,0))

textRect = text.get\_rect()

textRect.centerx = screen.get\_rect().centerx

textRect.centery = screen.get\_rect().centery+24

screen.blit(gameover, (0,0))

screen.blit(text, textRect)

else:

pygame.font.init()

font = pygame.font.Font(None, 24)

text = font.render("Accuracy: "+str(accuracy)+"%", True, (0,255,0))

textRect = text.get\_rect()

textRect.centerx = screen.get\_rect().centerx

textRect.centery = screen.get\_rect().centery+24

screen.blit(youwin, (0,0))

screen.blit(text, textRect)

while 1:

for event in pygame.event.get():

if event.type == pygame.QUIT:

pygame.quit()

exit(0)

pygame.display.flip()

# 六、调试与测试

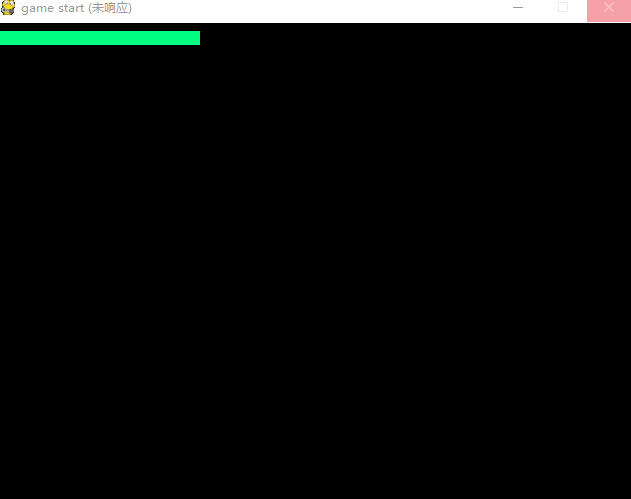
## 6.1调试方法

在调试这一方面，我主要使用Spyder进行调试，首先我添加断点，然后进入调试，到达断点后，单步调试，当把设置的断点调试结束后，停止调试。

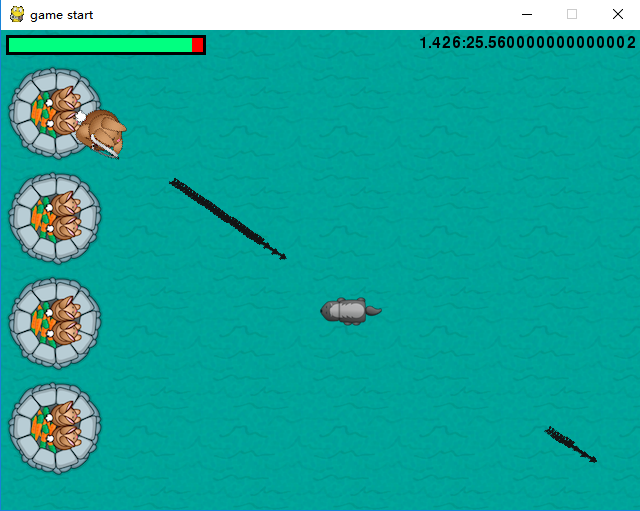
## 6.2测试结果的分析

经过测试，成功显示了生命条，生命值也被填充为绿色，然后实现了獾与城堡撞击之后獾消失的同时，城堡的生命值减少，同时，在獾向城堡冲过来的时候，也实现了箭头射中獾。当玩家技术很好的时候，玩家成功的玩到了90s的时候，显示出胜利的图片，当玩家技术不是很好的时候，玩家没有玩到90s生命值就减为0，显示出失败的图片。

经过测试，已实现了生命条的添加，獾与城堡和箭头的撞击，时钟的添加以及游戏结束的判定。



6-1生命值显示



6-2獾与箭头的碰撞

### 6.3问题及解决方法

在调试獾与城堡和箭头撞击那部分代码的时候，出现了badguying没有被定义的错误，经过一番查看，我发现我没有加载獾的图片，以至于出现了这个错误，之后我把加载图片这一部分加上去，就可以成功运行了。

在添加时钟和游戏结束判断有一个字体的设置，起初我设置的是‘arial’，但是调试的时候显示，没有这个字体，之后我就改为None,python默认字体，然后成功运行。

# 七、设计总结

课程设计结束了，在这次的课程设计我们通过小组合作完成了一个游戏的设计，总的来说，我们这次课设完成了刚开始我们对整个游戏的设想，我也成功的完成了我们组长分给我的任务，不得不说，过程是很充实的。在这次课设，在完成游戏结束判断这一块的时候，起初我只是在游戏结束的时候显示出是胜利还是失败的图片。之后，我想到之前玩的游戏在结束的页面都会有再来一局和游戏结束这两个选项，然后，我就根据这个想法，收集相关的资料，想在显示页面实现这个功能。我在修改的时候，发现我没有办法实现调用之前的代码，最后也就没有实现这一功能，所以，当玩家要在玩一次这个游戏就要先退出这个页面，再重新运行代码，在这一点上，我觉得还需要很大的改进。

对于这次课设，虽然说只有短短的几天，但还是有很大意义的，通过这次课设，让我对pygame有了一个了解，作为游戏开发的第三方库，pygame确实提供了很大的帮助。在这几天和队友共同谈论课题，然后根据题目选择自己适合的模块进行设计，最后将这些模块组装起来，当设计的游戏最后成功的运行，那种自豪感油然而生。