

## CS159 程序设计课程

### “我爱学习，学习使我快乐” 小游戏课题报告

小组名称： 300.75

小组成员： 邵心怡，曾心荷，杨煊

指导老师： 鲍杨

2019.06.09

## 一、课题背景

我们所创建的小游戏程序灵感来源于知名游戏《植物大战僵尸》。《植物大战僵尸》是由 PopCap Games 开发的一款益智策略类单机游戏，玩家通过武装多种植物切换不同的功能，快速有效地把僵尸阻挡在入侵的道路上。结合大学生的日常学习生活，我们用 Python 的 pygame 写出一个小游戏，在游戏中玩家利用经管类学生所学习的课程（如数分，经原，思修）来抵御娱乐项目（如小说，游戏）的入侵，以表达鼓励大家抵抗娱乐的诱惑好好学习的主题。

## 二、游戏介绍

我们用 pygame 建立了一个 960\*600 像素的窗口，游戏开始时会播放音乐《mathematics》。

游戏界面的左边是一张“我爱学习，学习使我快乐”的图片，一旦这个区域被娱乐项目入侵，那么游戏结束，弹出一张“退学”的图片。

在游戏过程中，娱乐项目会随机分布在各个轨道从右边进入，玩家通过拖拽各类课程（以书籍图片展示）来抵御娱乐诱惑。玩家需要“种植”DDL 并收集其产生的压力来获取学习的动力，从而拥有更多课程。其中数分和经济学原理会分别发射积分与曲线攻击娱乐使其渐渐消失（由于数分很难，它攻击的速度将是经济学原理的两倍），思修可以与一个娱乐项目同归于尽，体现其期末复习时爆发式的杀伤性。

## 三、问题的分析与解答

### 1. 游戏窗口

（1）建立一个窗口来进行游戏

```
#弹出窗口，设置屏幕宽高
pygame.init()
size = (960, 600)
screen = pygame.display.set_mode(size)
```

在 import pygame 之后，我们对其进行初始化，用 xxx 指令弹出制定宽高的屏幕

（2）为了增强可玩性，弹出窗口后需要播放音乐

```
#加载音乐
pygame.mixer.init()
pygame.mixer.music.load("material/music.mp3")
pygame.mixer.music.play(-1,0)
```

pygame 的 mixer 模块包含用于加载 Sound 对象和控制播放的类，我们对其进行初始化后倒入本地音乐，play 指令中的 -1 使得这首歌可以循环播放

（3）一开始的游戏界面应该包括背景图案，初始的金币，购买栏目三个板块

#屏幕各个区域的图片的载入，重新设置他们的像素

```
backgroundImg = pygame.image.load('material/images/background.jpg').convert_alpha()
backgroundImg=pygame.transform.scale(backgroundImg,(960,600))
DDLImg = pygame.image.load('material/images/DDL.png').convert_alpha()
DDLImg=pygame.transform.scale(DDLImg,(60,80))
jingyuanImg = pygame.image.load('material/images/jingyuan.jpg').convert_alpha()
jingyuanImg=pygame.transform.scale(jingyuanImg,(60,80))
```

首先我们用 pygame.image 的自带方法导入本地图片，重新改变他们的像素大小

#显示金钱栏的分数

```
score = 500
myfont = pygame.font.SysFont('arial', 30)
txtImg = myfont.render(str(score), True, (0, 0, 0))
```

设置金钱栏分数的字体名称和字体大小。之后通过 render() 来生成 Surface 对象了，再用 blit() 显示在屏幕上。(0,0,0) 说明字体颜色是黑色

# 确定各个图片的位置

```
screen.blit(DDLImg, (330, 10))
screen.blit(jingyuanImg, (400, 10))
screen.blit(shufenImg, (470, 10))
```

用 blit()显示图片，第二个参数是具体位置

## 2. 获取课程

(1) 上放购买栏点击相应的图标后，鼠标上会有一个相应的课程跟随它移动

# 选择卡片

```
if 330 <= x <= 380 and 10 <= y <= 80 and int(score) >= 50:
    choose = 1
elif 400<= x <= 450 and 10 <= y <= 80 and int(score) >= 50:
    choose = 2
```

# 拖动卡片

```
(x, y) = pygame.mouse.get_pos()
if choose == 1:
    screen.blit(DDLImg, (x - DDLImg.get_rect().width // 2, y - DDLImg.get_rect().height // 2))
--
```

获取鼠标点击的位置 (x, y)，通过鼠标点击的位置来判断它点在哪一张图片上，如果金钱数量足够，那么显示相应的图片，并通过坐标变换使得鼠标点在图片中心

(2) 为数分，思修，DDL，经原分别设置不同的价格，拖拽动作完成之后，会在金钱栏目相应扣除。

# 花钱

```
score -= 50
myfont = pygame.font.SysFont('arial', 30)
txtImg = myfont.render(str(score), True, (0, 0, 0))
```

更新 score 的值，并进行显示

(3) 鼠标按住左键进行拖拽。只需要拖拽到一个大致的位置松开鼠标就可以让课程自行到达指定位置

```
if choose == 1:
    trueX = x // 100 * 100
    trueY = y // 120 * 120
    canHold = True
```

$x//100*100$  对坐标进行处理，图片可以自行到达规范的位置

### 3.产生金币

(1) 种植的 DDL 会产生“压力”图案

```
for event in pygame.event.get():
    if event.type == GENERATOR_STRESS_EVENT:
        if len(ddlList) > 0:
            timeNow = time.time()
            for ddl in ddlList:
                if timeNow - ddl.lasttime >= 5:
                    ddl.lasttime = timeNow
                    stress = Stress(ddl.rect)
                    stressList.add(stress)
```

如果 DDL 的个数大于 0，那么对于每个 DDL 会重复生成 stress，stress 的初始值的位置由 ddl 的位置而定

```
class Stress(pygame.sprite.Sprite):
    def __init__(self, rect):
        super(Stress, self).__init__()
        self.image = pygame.image.load('material/images/stress.png').convert_alpha()
        self.image = pygame.transform.scale(self.image, (40, 40))
        self.rect = self.image.get_rect()
        self.rect.top = rect.top - 30
        self.rect.left = rect.left - 30
```

这里姑且顺使用 stress 的类来说明游戏的各种类，类中有 rect 参数（根据前一张图，其实是 DDL 的位置），再通过最后两行代码使得 stress 在 ddl 的左上角

(2) 通过点击“压力”对其进行采集，金币栏分数+50

```
for stress in stressList:
    if stress.rect.collidepoint((x, y)):
        stress.is_click = True
        score = int(score) + 50
        myfont = pygame.font.SysFont('arial', 30)
        txtImg = myfont.render(str(score), True, (0, 0, 0))
```

rect.collidepoint()检测一个点是否包含在该 Rect 对象内。由此判断鼠标是否点到了太阳  
如果颠倒了，更新分数

### (3) 压力图标分数的位置飞去并逐渐变小

```
#收集太阳
if self.is_click:
    self.rect.left -= (self.rect.left - 250) / self.times
    self.rect.top -= (self.rect.top - 30) / self.times

    self.image = pygame.transform.smoothscale(self.image,
        (self.rect.width // self.times * self.scale, self.rect.height // self.times * self.scale))
    if self.scale > 1:
        self.scale -= 1
    if self.rect.left <= 250 and self.rect.top <= 30:
        self.kill()
```

当鼠标点击之后，到左边和顶部的距离每隔一会变成原来的 4/5，大小也变小  
当 stress 到左上角的分数位置后，消失

## 4. 学科的攻击行为

### (1) 发射积分/曲线的频率

```
GENERATOR_JINGYUAN_EVENT = pygame.USEREVENT + 3
pygame.time.set_timer(GENERATOR_JINGYUAN_EVENT, 2000)
```

```
GENERATOR_SHUFEN_EVENT = pygame.USEREVENT + 4
pygame.time.set_timer(GENERATOR_SHUFEN_EVENT, 1000)
```

pygame.time.set\_timer()在事件队列上重复创建事件，在每给定的毫秒数上出现在事件队列上，  
等于后面的值需要不同来区别不同事件

后面一个参数决定了重复的事件，所以数分发射的较快

### (2) 当积分/曲线遇到娱乐项目后，后者生命值减少

```
for quxian in quxianList:
    for playing in playingList:
        if pygame.sprite.collide_rect_ratio(0.5)(quxian, playing):
            playing.energy -= 0.5
            quxianList.remove(quxian)
```

Sprite.collide\_rect\_ratio()是两个精灵之间冲突的检测，通过调整参数缩小检测范围  
当碰撞发射之后，通过 remove 方法使得曲线的图案消失，更新娱乐项目的生命值

### (3) 思修同归于尽的炸弹效果

和前者相同，用 Sprite.collide\_rect\_ratio()函数，但是 remove 之后对思修用 kill 函数使他消失，减少的生命大于娱乐项目本身的生命，使他直接爆炸

## 5. 娱乐项目

### (1) 随机分布在四个轨道

用随机函数，到顶部的举例 self.rect.top = 120+ random.randrange(0, 4) \* 120

(2) 生命值大于 0 时不断向左

```
if self.energy > 0:
    if self.rect.left > 0:
        self.rect.left -= self.speed
```

在存活的情况下，到左边的距离减去他的速度进行更新

(3) 在生命值减少为 0 后停止运动，显示爆炸画面，最终消失

```
if self.dietimes > 9:
    if self.dietimes > 24:
        self.kill()
    else:
        self.isAlive = False
        self.dietimes += 1
else:
    self.image = self.dieimages
    self.dietimes += 1
```

在 energy 为负的情况下，先显示爆炸图案，一段时间后消失

## 6. 游戏结束

(1) 当娱乐项目入侵左侧区域时，弹出退学

```
if playing.rect.left < 70:
    screen.fill([0, 0, 0])
    endImg = pygame.image.load('material/images/end.jpg').convert_alpha()
    endImg = pygame.transform.scale(endImg, (400, 300))
    screen.blit(endImg, (200, 150))
```

(2) 关闭弹窗

游戏结束关闭：

```
if playing.rect.left < 10:
    sys.exit()
```

点击×关闭：

```
if event.type == QUIT:
    pygame.quit()
    sys.exit()
```

## 四、有待改进的地方

1. 攻击击中目标的时候没有相应的音效，游戏体验受到影响。
2. 导入的“植物”和“僵尸”不会动，只是单纯的图片，略显单调。  
问题可以通过帧频来解决但是始终没有做出好的效果
3. “僵尸”走到“植物”面前时不会停顿并且对植物发动攻击，而是直接穿了过去。  
问题可以通过设立另外一个 collide 来解决，但是因为没动画效果会显得很奇怪

## 五、总结

通过这次共同合作一个小游戏，我们对 python 语言有了更加深刻的认识。虽然 debug 的时候遇到了很多的困难，也一度觉得难以继续进行下去，但是最后我们都克服了（当然是非常艰难地）。想到很好的点子的时候，我们也感到很兴奋，其中大有乐趣。很高兴我们终于完成了这个大作业。

## 六、致谢

我们要感谢鲍杨老师一个学期以来的付出，老师在指导我们的时候十分耐心，我们小组的同学一致认为老师非常可爱！