

一、实验目的与要求（200 字，说明要解决的问题）

本实验目的在于学习使用 Pygame 外部库的基本功能, 并利用所学习到的游戏开发框架以及事件队列处理机制, 利用 Pygame 库所含有的功能编写一个贪吃蛇小游戏。实现贪吃蛇小游戏的基本功能: 当贪吃蛇在吃到果实的时候, 能增加自身长度, 在碰到四周的时候结束游戏, 显示 GAME OVER; 贪吃蛇的移动可以通过键盘的 up,down,left,right 或者 wasd 四个按键控制; 随机产生果实, 产生的果实不能出现在贪吃蛇的内部; 增加小游戏的色彩元素, 实现贪吃蛇在移动过程中不断变化身体的色彩。

二、实验环境（200 字，列表格说明使用的外部库及其功能）

Pygame 是 python 中用于开发小游戏的一个外部库, 它提供了一个非常优良的开发 2D 游戏, 以及部分 3 维游戏的开发库。便于游戏开发入门, 了解游戏开发框架, 以及开发一些简单的 2D 小游戏。

编程环境	Pycharm	
调试平台	IDLE	
外部库	Pygame 库	
功能	屏幕绘制机制	控制屏幕尺寸, 模式 (全屏, 可伸缩等)
	pygame.display ()	修改窗口标题和图标
		窗口感知和刷新
	事件处理机制	可以控制处理系统, 键盘, 鼠标, 游戏杆等多种事件
	pygame.event ()	处理事件
		操作事件队列
		生成事件
	色彩与绘制机制	Pygame 色彩机制
		Pygame 图像绘制机制

三、实验内容（300 字，不含代码）

本实验利用 python 的外部库 Pygame, 实现了贪吃蛇小游戏的复刻。利用 Pygame

的部分功能函数，初始化了游戏图像化界面。实现了自定义的窗口标题，窗口图标，可适应的窗口大小。

游戏实现的原理和普通的程序代码有着不太一样的实现逻辑，实现游戏框架的搭建中更需要关注事件队列，然后对每一件发生的事件进行捕捉处理。对于不同的情况，编写对应的响应和调整方法。它的思维模式会更关注和计算机外部设备和一些硬件的交互。本实验内容使用小游戏的最小构建框架，并在其中添加需要的响应函数，实现游戏功能。

在编写游戏逻辑的时候，将整个游戏界面视为一个二维的网格。而蛇的头部和身体都是由像素点构成的，当蛇头遇到产生的食物（即 (x,y) 处的果实），则编写函数进行相应的相应。判断游戏胜负功能。

此外，本项目还引入了 Pygame 中的 color 设置方式，通过随机种子实现了贪吃蛇的移动变色功能。

四、实验设计与实现（300 字，不含代码）

本实验的重点分为四个部分：

1. 蛇身体和果实的生成：

我们将整个游戏场景视为一个二维的像素点格网，每一个格子都可以设置为一个像素点。实现蛇的身体和果实，我们只要通过一定的方式对于需要的格子进行涂色操作即可以实现蛇身体和果实的生成。

故编写了一个函数能够返回所希望的像素点位置，之后对其进行着色。

果实则利用随机函数生成。

2. 贪吃蛇吃到食物，身体变化的响应过程：

我们判断当蛇头的坐标和食物所在的坐标相同的时候，进入修改蛇身体长度的程

序,通过蛇现在身体运动的方向对于相应方向的像素格进行增删。通过时事判断,再染色的方式实现蛇的身度的改变。

3. 贪吃蛇身体颜色的动态变化

利用 Pygame 中可以对于不同色彩通道的 rgb 值,增加和蛇身体坐标相关的时刻变化的种子或者是 random 产生的种子。同时使得每次产生的 rgb 值均保持在 0-255 之间,实现蛇身体能够随着蛇的运动产生不同颜色的变化。

五、测试用例

可以通过 UP,DOWN,LEFT,RIGHT 按键或者 WASD 按键控制贪吃蛇运动,体验复刻版的贪吃蛇小游戏

六、收获与体会 (>100 字)

这次实现贪吃蛇复刻版小游戏让我收获颇丰。不仅是能够通过自己所学习的内容实现一个童年很喜欢的游戏所带来的的收获感,更是接触到了游戏构架的思维模式。在实现这个小游戏之前,我在慕课上简单了学习了有关于 Pygame 库的入门教程,了解了游戏的最小框架。知道了一系列的事件响应机制,这其中通过接受键盘的操作,利用事件队列进行判断的处理方式,让我觉得很新颖。与其他课程中所接触到的编程模式并不相同,也终于解开了一部分我对于计算机如何接受外部设备,处理外部设备所做的操作方式的疑惑。整个学习的过程中,我从一个对于游戏开发完全不懂的小白,到能够自己实现一个想完成的小游戏,其中的学习过程,也是我自身知识丰富的过程。

七、核心代码 (<100 行)

随机生成食物,且食物不能生成再蛇的身体内部

```
def get_food():
    while 1:
        position = Vex(row=random.randint(0, ROW - 1), col=random.randint(0, COL - 1))
        inBody = False
        # 如果生成的食物在蛇的身体里面,则重新生成
        if snake_head.row == position.row and snake_head.col == position.col:
            inBody = True
        for body in snake:
            if body.row == position.row and body.col == position.col:
                inBody = True
                break
        # 直到生成没有在蛇身体内的食物返回食物位置
        if not inBody:
            break
    return position
```

设置出口判断变量

```
quit = True
while quit:
    now = time.ctime()
    clock = now[11:19]
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            sys.exit()
        elif event.type == pygame.KEYDOWN:
            # 贪吃蛇不能够直接左右上下,要判断当前是在什么状态下前行
            if event.key == pygame.K_UP or event.key == 119:
                if direct == 'left' or direct == 'right':
                    direct = 'top'
            if event.key == pygame.K_DOWN or event.key == 115:
                if direct == 'left' or direct == 'right':
                    direct = 'bottom'
            if event.key == pygame.K_LEFT or event.key == 97:
                if direct == 'top' or direct == 'bottom':
                    direct = 'left'
            if event.key == pygame.K_RIGHT or event.key == 100:
                if direct == 'top' or direct == 'bottom':
                    direct = 'right'
```

对吃了果子的蛇身进行修改

```
eat:
    # 随机产生果子
    s_food = Vex(row=random.randint(0, ROW - 1), col=random.randint(0, COL - 1))
snake.insert(0, snake_head.copy())
not eat:
    snake.pop()

direct == 'left':
    snake_head.col -= 1
direct == 'right':
    snake_head.col += 1
direct == 'top':
    snake_head.row -= 1
direct == 'bottom':
    snake_head.row += 1
ad = False
snake_head.col < 0 or snake_head.row < 0 or snake_head.col >= COL or snake_head.row >= ROW:
    dead = True
for body in snake:
    if snake_head.col == body.col and snake_head == body.row:
        dead = True
        break
dead:
    print("GAME OVER")
    quit = False
```