

# I wanna be the pythonic guy

## ——Python 暑校大作业实习报告

作者：赵一鸣（组长） 王功臣 张漫溪 沈洛羽

**【摘要】** I Wanna Be The Guy 是早期难度较高的单机冒险游戏，在游戏中玩家将体验一个年轻、懵懂的孩子为了追寻理想中的角色所经历的过程，剧情变幻莫测，令人沮丧的跳跃和射击的挑战会使你对这个主角小孩产生敬意。此次分组大作业中，本小组主要利用 pgzero 来打造类似 I wanna 的动作冒险小游戏，通过大胆想象，设计精妙又充满陷阱的关卡，小组的每一个人都在实践中提升了自己的 python 编程技巧，并在这一过程中找到了独特的乐趣。

### 一、 选题及创意介绍

I Wanna Be The Guy 是很早的难度高到变态的单机游戏，这款游戏让玩家玩到手麻，一般来说不死上那么几千次是过不了的。在美国早期游戏业比较宁



静的时代，I Wanna Be The Guy 作为一款难度很高的冒险类游戏，没玩过的人绝对会成为冷嘲热讽的对象。在本次大作业中，我们设计了不同难度的关卡，玩家需要通过移动跳跃，躲过不同的陷阱，最终抵达下一关，如若未躲过陷阱，则须回到原点，再次挑战。在利用 pgzero 编写程序，实现类似原游戏中的各种操作与事件交互的基础上，我们可以根据自己的想法来自由构图、设计意想不到的关卡与“阴险”的陷阱，整个过程富有挑战性以及趣味性。

## 二、 设计方案

### 1. 项目整体方案:

为了让每一个成员都能够实践 python 游戏编程, 我们各自在主程序框架的基础上平行设计了自己独一无二的关卡。每个人的独立关卡设计完毕后, 我们进行了总体的整合, 将所有分程序的主要函数抽取后在主程序中关键的位置进行调用, 再经过适当的调整最终形成了一个完整的主游戏文件 iwanna.py(代码量 1400+lines)。为了方便玩家游戏, 组长克服重重阻碍, 贴心地用 pyinstaller 将整个项目打包, 最终的游戏文件夹包括一系列配置文件与核心可执行文件 iwanna.exe, 玩家不需要安装 python, 只需点击快捷方式即可运行。

i wanna	2020/7/11 11:49	快捷方式	1 KB
files	2020/7/11 11:44	文件夹	

图一: 成品展示

### 2. 主程序框架

#### 1) 全局变量声明

在程序开头声明游戏中所有可交互对象, 并使用列表结构储存所有对象, 方便操作与管理 (如右图)。

#### 2) 重置函数 reset ()

用于初始化或重置所有对象。在关卡开始与玩家重生的时刻, 这一函数保证地图上所有对象都在合理的位置上, 并保证一些全局控制变量有着正确的取值。

```
#全局设置
WIDTH = 1280
HEIGHT = 720
BOTTOM=710

#地图地面
bottoms=[]
#升降平台
platforms=[]
#背景
backs=[]
#存档点
saves=[]
#按钮
buttons=[]
#树木
trees=[]
#苹果
apples=[]
#尖刺
spines=[]
```

图二: 部分全局变量

#### 3) 绘图函数 draw ()

以一定帧速率绘制地图上所有可显示对象, 包括玩家、可交互物体、文字等。

#### 4) 更新函数 update ()

程序的主要函数, 默认在 draw 之前调用。用于控制人物与物体的互动, 触发各种事件等。

## 5) 按键控制函数 on\_key\_down() & on\_key\_up()

这两个函数主要处理用户的键盘行为, 包括左, 右, 空格, 保存, 重生。

## 三、 实现方案及代码分析

### 1. 多关卡实现:

由于整个程序包含多个独立的关卡, 这就意味着不同的布局与交互关系。为了能在不同关卡间跳转, 我们对主程序中 reset () 与 update () 两个函数做了分支处理, 再用一个全局变量记录当前所在关卡, 如此一来较为简洁地实现了多关卡游戏的目的。

```
if current_window==0:zymupdate()  
elif current_window==1:smyupdate()  
elif current_window==2:zymupdate()  
elif current_window==3:zmxupdate()  
elif current_window==4:wgcupdate()
```

```
def reset():  
    if current_window==1:smyreset()  
    elif current_window==2:zymreset()  
    elif current_window==3:zmxreset()  
    elif current_window==4:wgcreset()  
    elif current_window==5:ending()
```

图三: 多关卡处理 (具体见源码)

### 2. 物理引擎:

作为一个动作冒险类游戏, 物理引擎 (动作处理) 是必不可少的。在本项目中, 我们将物理引擎集成到了函数 update 中:

```
#运动模块  
player.vy+=player.ay  
player.vx=player.staticvx
```

```
#运动  
player.left+=player.vx  
if player.bottom<=1000:  
    player.bottom+=player.vy
```

图四: 部分处理运动的语句 (具体见源码)

为了尽量真实地模拟人物的移动与跳跃, 本程序特地引入了垂直方向上的重力加速度与水平方向的惯性参量, 通过适当调参以处理人物在整个地图内的移动。

### 3. 边界检测与碰撞处理:

由于 pgzero 统一以 “贴图” 的方式载入对象, 那么如何通过算法使载入其中的内容成为真正可交互的 “物体” 就是本程序设计的一大难点。在这

个项目中，我们使用边界检测的方式解决了这一问题。具体来说，就是通过特定算法代码块来限制人物的运动，形成“地面”，“墙壁”等概念，避免“穿模”等问题。碰撞方面，由于程序的限制，所有图片都被视为有着矩形轮廓，那么对于“尖刺陷阱”这类可交互对象不能简单地使用 colliderect 函数处理碰撞，而应该先对实物进行一定密度的边缘采样，再使用 colliderpoint 函数进行碰撞判断与交互：

```
def edge_sample():
    '''边缘采样函数，用于勾勒尖刺边缘以进行碰撞检测'''
    level=4#采样等级，数值越小越容易发生碰撞，游戏难度越高
    for spine in spines:
        if spine.image=='spine_up':
            x,y=spine.bottomleft
            x+=level
            for i in range(100):
                x+=(spine.width-level*2)/200
                y=(spine.height-level)/100
                spine.points.append((x,y))
            for i in range(100):
                x+=(spine.width-level*2)/200
                y+=(spine.height-level)/100
                spine.points.append((x,y))
        elif spine.image=='spine_down':
            x,y=spine.topleft
            x+=level
            for i in range(100):
                x+=(spine.width-level*2)/200
                y+=(spine.height-level)/100
                spine.points.append((x,y))
            for i in range(100):
                x+=(spine.width-level*2)/200
                y+=(spine.height-level)/100
                spine.points.append((x,y))
```

图五：边缘采样函数（具体见源码）

## 4. 陷阱

众所周知，“陷阱”无数一直都是 I wanna 系列游戏最显著的特点，而我们设计的游戏也不例外。我们通过巧妙运用 animate 函数构造许多内含特殊响应机制的动画来让游戏充满各式各样的生动的陷阱，增强了趣味性。

四、 后续工作展望

- 1) 调整各关卡的游戏的难度，使其过渡性更加合理，更加具有递进性。
- 2) 适当加强游戏的复杂度以及难度，每个关卡可适当增加陷阱与更多不同种类的可交互对象，加强游戏的互动性及体验感。
- 3) 增加游戏关卡，可根据不同的难度系数将游戏分类，以此吸引不同类型的玩家；例如：简易型吸引低端玩家打发时间，困难型吸引高端玩家进行挑战。
- 4) 完善存档系统，使玩家记录可以被永远保留，防止因丢失存档而导致的崩溃。

五、 小组分工合作

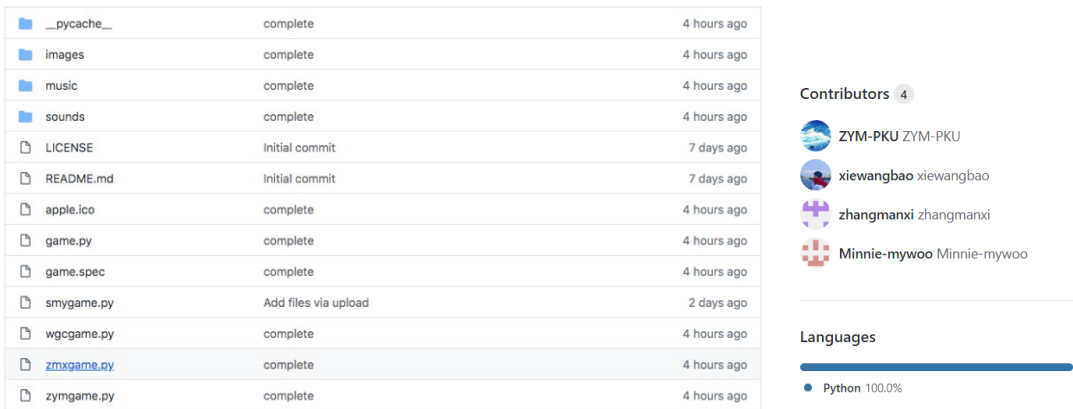
1. 任务分工

- 1) 每人首先尝试设计了自己独一无二的关卡，最后组长根据关卡难度将其进行合并及调整，加入开头和结尾，整合成一个完整的游戏。
- 2) 此外，张漫溪同学完成了海报制作，王功臣同学完成了短视频的制作，沈洛羽同学与组长赵一鸣同学一起完成了本实习报告的撰写与校对。

2. 使用平台与小组讨论

1) GitHub

我们在 GitHub 平台创建了一个小组项目：**PythonSummerSchool**,



图六：github 上的项目

将编写程序所需要的游戏背景，人物等各种图片，音频，以及各自编写的源代码等资料上传至该平台，方便组内信息共享，提高了合作效率。

项目地址：<https://github.com/ZYM-PKU/PythonSummerSchool>

## 2) 微信群讨论

我们通过微信群进行任务的布置，传输文件与传递重要信息，提出各自的创意与意见建议，讨论工作最新进展。



图七：微信群内的小组讨论

## 六、 总结

通过高效分工与通力合作，我们小组最终使用 python 做出了一个冒险小游戏，成功完成了本次大作业的任务要求。在这一过程中，每一名成员都锻炼了 python 程序编写能力，提高了合作学习的水平，并在游戏的设计中找到了自己的乐趣。

Ps：一切都在游戏里了，想要了解我们的成果，不妨去玩一玩吧，说不定会发现彩蛋哦！