

最近用python的tkinter写了一个极简版的植物大战僵尸，目前完成度差不多了。在这个版本里，大家可以非常简单随意地修改所有植物和僵尸的游戏参数，可以自己定制关卡，可以自己制作新植物，新僵尸，而且所有游戏画面里的东西都可以改，也包括地图，背景音乐和所有游戏音效，如何修改我接下来会说。我算是从零开始写了一个pvz，一切游戏布局和僵尸和植物的游戏逻辑算法，全部的游戏界面设计，数据结构都是自己原创，没有参考任何源代码。目前还没有把所有的植物和僵尸的算法都写完，不过很快的。

左上角是当前得到的阳光，右上角是天上落下的阳光，因为是极简版，所以玩家直接在右上角出现阳光的时候点击就可以获得阳光。草地的横向纵向格子数量都可以自己修改。非常值得一提的是，这个极简版和原版相比，多了屏幕下方的一个当前状态显示栏，可以显示你当前正在做什么事情（比如在哪里种了什么植物，哪个格子上有没有植物，有什么植物），你当前哪个位置的植物被吃掉了，你的阳光够不够一个植物的种植，这个植物是否还在冷却等等。这个状态栏会让玩家更清楚自己正在玩的游戏的整体形势。另一个不一样的地方是这个状态栏右边会显示当前你杀死的僵尸数量。

正下方的僵尸进度条和原版设计得差不多，在你定制自己的关卡时，这个进度条也会相应做出改变。

目前的版本的游戏结构设计是给每个植物和每个僵尸一个独立的脚本文件，并且所有植物的脚本文件都放在

plant_scripts文件夹里，所有僵尸的脚本文件都放在zombie_scripts文件夹里。

每一个关卡也都对应着一个独立的脚本文件，所有关卡的脚本文件都放在stages文件夹里。

resource文件夹是存放游戏所有的图片，音乐，音效的文件夹，也就是游戏的资源文件夹，如果有需要批量修改参数的脚本

也需要放在这里，然后在pvz_config配置文件里把modified_file修改为脚本文件名即可。

在resource文件夹里还有一个common.py的文件，这个是在读取关卡时，会转到resource文件夹，这个common.py的脚本文件

会进行每种僵尸的音效资源的预处理。

接下来我会详细地说明这个极简版pvz的数据结构设计。想要自己制作新植物和新僵尸的人请务必要认真看完。

首先是植物类型的设计。每一种植物都属于植物类型。植物类型的内部参数如下：

```
name,  
img,  
price,  
hp,  
cooling_time,  
hp_img,  
attack_interval,  
bullet_img,  
bullet_speed,  
bullet_attack,  
bullet_sound,  
sound_volume,
```

self_attack,
change_mode,
func,
bullet_func,
effects,
no_cooling_start,
is_bullet,
dead_normal,
img_transparent,
rows,
columns

我接下来——讲解这些参数。

name是植物的名字，比如豌豆射手，那么name就是"豌豆射手"。

img是植物的图片文件路径，也就是植物在游戏里出现的样子，包括卡牌上和种植之后。

price是植物的阳光价格。

hp是植物的生命值。

cooling_time是植物的冷却时间，单位为秒。

hp_img是如果一个植物的生命值减少到一定量后，图片会发生变化，那么就设置为

((生命值剩下的百分比1, 对应的图片路径1), (生命值剩下的百分比2, 对应的图片路径2), ...)

这里的百分比没有乘上100，比如坚果剩下三分之二，也就是66%的血的时候会变成被啃的图片，假如这个图片路径在"resource/bite.png"，那么这里就写((2/3, "resource/bite.png"),)。

假如有个植物剩下一半血的时候会变成half.png这个图片，那么就是((0.5, "resource/half.png"),)。

attack_interval是植物的攻击间隔，单位为秒，比如豌豆射手每隔2秒发射一个豌豆，那么这里就是2。

bullet_img是植物发射的子弹的图片路径。

bullet_speed是植物发射的子弹的移动速度，单位为毫秒（千分之一秒）。比如豌豆射手发射的豌豆每过0.2秒移动一格，那么这里就是200。

bullet_attack是植物发射的子弹的攻击力。

这个极简版pvz的血量设计是以一个豌豆射手的一个豌豆的攻击力作为单位，也就是说，豌豆射手的豌豆攻击力为1，对应的普通僵尸的血量为10，因为在原版里一只普通僵尸满血状态下刚好可以被豌豆打10下。豌豆射手一般被僵尸啃5下没掉，因此豌豆射手的血量默认为5。

bullet_sound是植物发射子弹时的声音文件路径。

sound_volume是植物发射子弹的声音大小调整，1为100%的音量（最大音量），假如是0.6就是60%的音量。

self_attack为植物本身的攻击力（如果有的话），比如大嘴花自身就有很大的攻击力。

关于self_attack目前还没有写相应的算法，接下来写大嘴花时会加入到游戏的普遍算法中。

change_mode是之前的hp_img里的判断生命值的模式选择参数。change_mode默认为0。

change_mode为0的时候，血量小于或等于百分比就变成对应的图片。

change_mode为1的时候，血量小于或等于一个值时就变成对应的图片。

change_mode为2的时候，血量减少的量大于或等于一个值时就变成对应的图片，也就是被啃了多少下之后。

func是植物在游戏每次的循环里运行的算法，是一个带有两个参数的函数，第一个参数self是植物自己，第二个参数games是游戏主体，

通过这两个参数就可以实现植物和游戏主体的全面互动，玩家自己制作新植物的算法时也可以随心所欲。如果一个func不够用的话，

这个func自己可以调用其他的函数，只需要写在植物脚本里就行了。

bullet_func是游戏在玩家暂停之后，继续游戏的时候对之前暂停时停下的子弹的后续处理函数，一般为继续之前同样的子弹移动函数。

effects是一个字典，键为一个植物对其他的植物或子弹或僵尸会产生效果影响类型，对应的值为一个函数，可以把受到影响的植物或子弹或僵尸

的参数进行修改或者其他你想做的事情。会受到影响植物或者子弹或者僵尸自己的脚本里也需要写上对应的探测部分，一般来说写在func里即可。

no_cooling_start是一个布尔值，为True的时候，游戏开局时直接冷却好，后面正常冷却。为False的时候，游戏从开局时就正常冷却。默认值为False。

一般只有向日葵这个植物会用到这个参数，向日葵的no_cooling_start是True。

is_bullet是一个布尔值，为True的时候，植物的bullet_img，也就是子弹图片会按照地面图片的三分之一尺寸进行缩放，

为False的时候则不会进行缩放，按照原尺寸。默认值为True。

dead_normal是一个布尔值，为True的时候，植物的生命值小于或等于0的时候会被判定为死亡并且消失，为False的时候不会，

玩家可以在func里设置生命值小于或等于0时这个植物要做什么事情。默认值为True。（比如爆炸坚果这个植物就需要这个参数设置为False）

img_transparent是一个布尔值，为True的时候，植物的图片会被当成背景透明的图片，并且调整好尺寸和位置和地面图片合成作为新的植物图片，

为False的时候直接使用植物的图片而不作任何处理。默认值为False。

rows是植物所在的行数，columns是植物所在的列数，这两个值共同表示当前植物的位置。

以上就是植物的所有参数，这些参数全部都可以随意修改。在pvz_config.py这个文件里的plant_dict，有目前我写完的植物参数，这些都可以随便改。

接下来是僵尸类型的设计。每一种僵尸都属于僵尸类型。僵尸类型的内部参数如下：

name,

img,

hp,

move_speed,

attack,

attack_speed,

attack_sound,
dead_sound,
hit_sound,
hit_sound_ls,
hp_img,
start_func,
eachtime_func,
repause_func,
other_sound,
img_transparent,
rows,
columns,
appear_time,
change_mode
name是僵尸的名字。
img是僵尸的图片的文件路径。
hp是僵尸的生命值。生命值单位我之前在植物那边说过，以一个豌豆的攻击力为1，因此普通僵尸的生命值默认为10。
move_speed是僵尸的移动速度，多少秒移动一格（一整个草地的格子），比如一只普通僵尸每过9秒可以走过一整格草地，那么move_speed就设置为9。
attack是僵尸的攻击力，每次攻击对方的生命值减少多少。
attack_speed是僵尸的攻击速度。
attack_sound是僵尸攻击时的声音文件路径。
dead_sound是僵尸死亡时的声音文件路径。
hit_sound是僵尸被植物攻击时的声音文件路径。
hit_sound_ls是僵尸的生命值减少到多少之后，对应的声音文件路径，写法参考之前植物的hp_img。
hp_img，也一样参考之前植物的hp_img。
start_func是这个僵尸在游戏里出现的时候做的事情，玩家在写脚本时可以从regular里面import默认的start_func，regular里默认的start_func是zombie_move。
eachtime_func是这个僵尸在游戏每次循环都会做的事情，在regular里面默认为next_to_plants。
repause_func是这个僵尸在游戏暂停后继续游戏时会做的事情，在regular里面默认为repause。
other_sound是僵尸需要的除被攻击的声音，攻击的声音，被打败时的声音之外的其他音效。
img_transparent参考植物里的img_transparent。
rows和columns是僵尸的行数和列数。
appear_time是僵尸在游戏开始之后的出现时间，单位为秒，也就是僵尸在游戏开始之后多少秒会出现。

change_mode参考之前植物的change_mode。

到这里植物和僵尸的数据结构就差不多说完了。其实真正复杂的是不同的植物和僵尸的游戏逻辑算法实现，不过这样的数据结构设计，让我在设计算法时很感觉很流畅，条理也很清晰。

接下来先说下游戏中的控制。鼠标左键可以做和原版一模一样的事情，选择植物，种植植物，铲除植物，拿阳光等等，除此之外还有很多原版没有的功能，比如鼠标左键点击植物卡片时，下面的状态栏会写你选择了哪种植物，当你种植时，下面的状态栏会写你把什么植物种在了第几行第几列。当你直接点击一块草地时，下面的状态栏会显示当前的格子有没有植物，如果有，会显示上面有什么。当你铲除植物时，如果没有植物，下面的状态栏会嘲讽你一句（笑），如果有植物，状态栏会写你铲除了什么位置的什么植物。当你点击阳光时，状态栏会显示你获得了多少阳光。

鼠标右键可以取消当前选择的東西。比如你选择了一个植物要种植，点右键就可以取消。当你选择了铲子，点右键也可以取消。按键盘上的空格键可以暂停游戏，按P继续游戏。

游戏一开始的画面是让你选择植物，这里跟原版是一样的，直接点击植物卡片就会看到选到卡槽中，在卡槽里点植物就可以取消选择这个植物。

选择完植物后，点击开始游戏就可以开始。

所有的游戏参数在pvz_config.py这个文件里都可以修改，这里再讲一些其他的游戏参数。

background_music是游戏的背景音乐文件路径。

choose_plants_music是选择植物界面的背景音乐。

sunshine_img是阳光的图片文件路径。

还有其他非常多参数，比如你甚至可以改游戏的标题，图标，还有地图的图片也可以改，lawn_img那个就是地图的图片文件路径。

接下来说如何定制关卡。在这个版本里，除了植物类型和僵尸类型以外，还有一个关卡类型。关卡类型的初始化参数只有一个，就是总共有几个旗帜（也就是有几大波僵尸）。

现在我们新建一个关卡，名字为current_stage，因为主程序读取的关卡变量名就是current_stage，所以这里就固定写current_stage。这样可以新建一个有两个旗帜的关卡：

```
current_stage = Stage(2)
```

假如我们要在第一个旗帜到来之前，设置在1分钟内随机的行数出现30只僵尸，僵尸类型从普通僵尸和路障僵尸里面选，那么就写part1 = [get_zombies(random.choice(['普通僵尸', '路障僵尸']), random.randint(0, 4), 8, random.randint(1, 60)) for i in range(30)]。这里需要讲下get_zombies这个函数，get_zombies函数第一个参数是僵尸的名字，第二个参数是僵尸出现的行数，第三个参数是僵尸出现的列数，第4个参数是僵尸出现的时间（单位为秒）。值得注意的是，在当前part1这波里的僵尸全部被打完之前，下一波僵尸不会出现，这里也是跟原版一样的，因此这个出现的时间的起始参照点就是每一波僵尸开始的时候，比如到第二波开始后，就是按照从第二波开始的时间为开始时间。part1是一个有30只僵尸的列表，我们现在只需要current_stage.set_normal(0, part1)就可以把part1设置为第一个旗帜之前出现的僵尸。

如果是到了旗帜的时候，也就是“一大波僵尸要来袭了”的时候，我们也和part1一样自己定制僵尸，比如叫做wave_1，然后只需要current_stage.set_waves(0, wave_1)就可以把wave_1设置为第一个旗帜的时候出现的僵尸。

（不懂编程的人请注意，一般编程里都是以0作为第一个，因此0就代表第一波，1代表第二波，等等）

至于制作新植物和新僵尸，那你可能真的得会一点python了2333，或者其实我也在想，有没有办法做一个程序可以让用户很轻松地自己制定一个新植物和新僵尸的游戏算法，即使不会编程的人也会写的那种难度，然后我自己写个解释器翻译成python代码。这些算法可以保存为独立的文件然后在配置文件（pvz_config.py）里面用到就可以。比如我们想写一个新植物叫做苹果，这个苹果看似好吃但是却有毒，会让每一个吃到苹果的僵尸在接下来的15秒内每秒扣一些生命值，这个扣的量从1到5之间不等（随

机选择)。每个僵尸只会咬一口苹果，然后就直接走过去。这个算法只要会一点python，都很快就能写完，不过不会python的人，可能就不是那么容易设计了，如果我写个算法定制器，可能就很方便了。我接下来会试着来写一个。

如果当前的关卡打赢了，下面的状态栏会显示你赢了，然后过7秒后游戏自己关闭。僵尸进家的时候，下面的状态栏显示你输了，然后也是过7秒游戏自己关闭。

这个极简版的pvz里面目前可以自己制作关卡，然后打开游戏可以选择自己想要玩的关卡开始玩。我也为这个版本写了一些小游戏，目前写完了原版里的锤僵尸和我是僵尸（包括定制关卡和无尽模式）。在游戏里点击设置按钮可以调整当前的背景音乐的音量，音效的音量以及可以改变背景音乐。后续会加入更多可在游戏过程中设置的功能。