|  |
| --- |
|  |
| 作品说明书 |
| 秋秋贪食蛇 |

|  |
| --- |
| 19103670131 蔡秋霞 |

目录

[一、 项目内容 1](#_Toc74756950)

[二、 项目方案 1](#_Toc74756951)

[三、 技术要点 2](#_Toc74756952)

[四、 难点及解决过程 8](#_Toc74756953)

[五、 代码及运行结果 9](#_Toc74756954)

# 项目内容

本项目作品是一个Pygame小游戏——贪食蛇。本项目的贪食蛇除了能够简单地控制小蛇方向吃掉食物之外，还另外添加了三个道具，分别是生命药水、加速剂和毒药，以增加游戏的难度与趣味性。不同的道具有不同的用途、不同的出现时机和不同的消失时间。当小蛇吃到食物，与此同时，蛇身变长，行动速度加快。游戏进行过程中，实时显示用户当前得分。当小蛇死亡后，游戏结束，根据用户的得分给与用户一个评定等级，并提示用户按下键盘上空格键重新开始游戏。

# 项目方案

图 一 秋秋贪食蛇的设计框架

1. 相关设置：

里面包含了这个游戏所需要的固定设置，方便下面的函数调用。

1. 初始化：

初始化Pygame这个游戏库。

1. 游戏主体循环运行：
   1. 蛇、食物、道具的位置初始化：

对这些位置进行初始化。

* 1. 监听键盘、移动蛇位置：

通过对键盘的监听，控制蛇的运动。

* 1. 判断游戏状态：

如果触发死亡，则结束游戏；否则继续游戏。

* 1. 分数更新：

蛇的长度增加则分数更新。

* 1. 评定等级：

根据用户所得的分数评定等级。不同分数段获得不同等级。

* 1. 速度变化：

随着小蛇的生长，小蛇的运动速度也随之加快。当小蛇吃到加速剂时也会使小蛇的速度增加。

1. 结束信息：

触发死亡游戏结束时出现的界面，显示游戏结束的文本内容，重新开始游戏的提示内容以及用户所获得的等级的信息内容。

# 技术要点

1. 蛇、食物、道具类的初始化定义

定义蛇、食物、道具类以其所需的方法。

# 蛇类

# 点以25为单位

class Snake(object):

# 初始化各种需要的属性 [开始时默认向右/身体块x5]

def \_\_init\_\_(self):

self.dirction = pygame.K\_RIGHT

self.body = []

for x in range(5):

self.addnode()

# 无论何时 都在前端增加蛇块

def addnode(self):

left, top = (0, 0)

if self.body:

left, top = (self.body[0].left, self.body[0].top)

node = pygame.Rect(left, top, 25, 25)

if self.dirction == pygame.K\_LEFT:

node.left -= 25

elif self.dirction == pygame.K\_RIGHT:

node.left += 25

elif self.dirction == pygame.K\_UP:

node.top -= 25

elif self.dirction == pygame.K\_DOWN:

node.top += 25

self.body.insert(0, node)

# 在速度相反方向增加蛇块，使小蛇撞到自身而死亡

def addnode2(self):

left, top = (0, 0)

if self.body:

left, top = (self.body[0].left, self.body[0].top)

node = pygame.Rect(left, top, 25, 25)

if self.dirction == pygame.K\_LEFT:

node.left += 25

elif self.dirction == pygame.K\_RIGHT:

node.left -= 25

elif self.dirction == pygame.K\_UP:

node.top += 25

elif self.dirction == pygame.K\_DOWN:

node.top -= 25

self.body.insert(0, node)

# 删除最后一个块

def delnode(self):

self.body.pop()

# 死亡判断

def isdead(self):

# 撞墙

if self.body[0].x not in range(SCREEN\_X):

return True

if self.body[0].y not in range(SCREEN\_Y):

return True

# 撞自己

if self.body[0] in self.body[1:]:

return True

return False

# 移动！

def move(self):

self.addnode()

self.delnode()

# 改变方向 但是左右、上下不能被逆向改变

def changedirection(self, curkey):

LR = [pygame.K\_LEFT, pygame.K\_RIGHT]

UD = [pygame.K\_UP, pygame.K\_DOWN]

if curkey in LR + UD:

if (curkey in LR) and (self.dirction in LR):

return

if (curkey in UD) and (self.dirction in UD):

return

self.dirction = curkey

# 食物类

# 方法： 放置/移除

# 点以25为单位

class Food:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具

# 道具一： 生命药水类

# 道具用途：当玩家得分达到500时，出现该道具，当玩家吃到该道具时，得分加100

# 方法： 放置/移除

# 点以25为单位

class Life:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具二： 加速剂类

# 道具用途：得分加500，小蛇速度加快

# 方法： 放置/移除

# 点以25为单位

class Speed:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具三：毒药类

# 道具用途：得分减100，小蛇中毒死亡

# 方法：放置/移除

# 点以25为单位

class Poison:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

1. 显示文本的函数

# 显示文本

def show\_text(screen, pos, text, color, font\_bold=False, font\_size=60, font\_italic=False):

# 获取系统字体，并设置文字大小

cur\_font = pygame.font.SysFont("宋体", font\_size)

# 设置是否加粗属性

cur\_font.set\_bold(font\_bold)

# 设置是否斜体属性

cur\_font.set\_italic(font\_italic)

# 设置文字内容

text\_fmt = cur\_font.render(text, 1, color)

# 绘制文字

screen.blit(text\_fmt, pos)

1. 监听键盘、移动蛇位置

设置一个循环，监听键盘的按键是否按下，通过设定各按键对应的方向，控制蛇头的方向改变。

while True:

for event in pygame.event.get():

if event.type == pygame.QUIT:

sys.exit()

if event.type == pygame.KEYDOWN:

snake.changedirection(event.key)

# 死后按space重新

if event.key == pygame.K\_SPACE and isdead:

return main()

1. 画蛇身

# 画蛇身 / 每一步+1分

if not isdead:

scores += 1

snake.move()

for rect in snake.body:

pygame.draw.rect(screen, (20, 220, 39), rect, 0)

1. 食物处理与食物投递

# 食物处理 / 吃到+50分

# 当食物rect与蛇头重合,吃掉 -> Snake增加一个Node

if food.rect == snake.body[0]:

scores += 50

food.remove()

snake.addnode()

# 食物投递

food.set()

pygame.draw.rect(screen, (136, 0, 21), food.rect, 0)

1. 设置道具出现时机与用途

# 当scores达到500时，出现生命药水道具

if scores >= 500:

# 生命药水处理 / 吃到scores+100分

# 当生命药水道具rect与蛇头重合,吃掉 -> Snake增加一个Node

if life.rect == snake.body[0]:

scores += 100

life.remove()

snake.addnode()

# 生命药水投递

life.set()

pygame.draw.rect(screen, (238, 153, 221), life.rect, 0)

# 当1000 <= scores <= 1500 或者 2500 <= scores <= 3000,出现加速剂道具:

if 1000 <= scores <= 1500 or 2500 <= scores <= 3000:

# 加速剂类处理 / 吃到scores+500分 / 速度加5

# 当加速剂类道具rect与蛇头重合，吃掉 -> Snake增加一个Node

if speed.rect == snake.body[0]:

scores += 500

snakeSpeed += 5

speed.remove()

snake.addnode()

# 加速剂类投递

speed.set()

pygame.draw.rect(screen, (142, 68, 173), speed.rect, 0)

# 当蛇身大于10时，出现毒药道具：

if len(snake.body) > 10:

# 毒药处理 / 吃到 scores-100分

# 当毒药道具rect与蛇头重合，吃掉中毒死亡

if poison.rect == snake.body[0]:

scores -= 100

poison.remove()

snake.addnode2()

# 毒药投递

poison.set()

pygame.draw.rect(screen, (239, 246, 79), poison.rect, 0)

1. 控制小蛇速度

# 控制小蛇的速度，蛇身长度越长速度越快

if len(snake.body) < 10:

snakeSpeed = 5 + len(snake.body) / 2

else:

snakeSpeed = 16

# clock.tick(10)

clock.tick(snakeSpeed)

1. 实时显示得分

# 显示分数文字

show\_text(screen, (50, 700), 'Scores: ' + str(scores), (223, 223, 223))

1. 结束信息

# 显示死亡文字

isdead = snake.isdead()

if isdead:

show\_text(screen, (200, 300), 'YOU DEAD!', (227, 29, 18), False, 100)

show\_text(screen, (250, 380), 'press space to try again...', (0, 0, 22), False, 30)

# 根据得分评定等级

if scores <= 500:

show\_text(screen, (250, 450), 'Your level is F', (161, 163, 166), False, 60)

elif 500 < scores <= 1000:

show\_text(screen, (250, 450), 'Your level is D', (0, 174, 157), False, 60)

elif 1000 < scores <= 1500:

show\_text(screen, (250, 450), 'Your level is C', (0, 154, 214), False, 60)

elif 1500 < scores <= 2000:

show\_text(screen, (250, 450), 'Your level is B', (253, 185, 51), False, 60)

else:

show\_text(screen, (250, 450), 'Your level is A', (247, 172, 188), False, 60)

# 难点及解决过程

1. **难点：**生命药水道具的用途设置：原本预期功能是当小蛇吃到生命药水后，可以增加一条生命，即生命值加一。当小蛇撞墙或者撞到自己时，生命值减一，若生命值为零时，小蛇死亡，游戏结束；否则初始化小蛇，恢复小蛇原长，而分数继续累加，游戏继续。

**解决过程：**已知小蛇原长为5，当小蛇生命值减一但不为零时，利用delnode（）函数删除小蛇最后一个块直到小蛇长度为5，并将删除的块颜色改为背景色。但未得到解决的是，删除到原长时，已经删除的块的后面又多出一块。另一方面，小蛇撞墙或撞到自己后，位置如何回到初始位置，也未得到解决方案。因此本项目中的生命药水，唯一的用途只有附加分数100。

1. **难点：**加速剂如何使小蛇的速度加快：当小蛇的蛇头与加速剂道具重合时，如何实时使小蛇的速度加快？

**解决过程：**当小蛇的蛇头与加速剂道具重合时，通过改变snakeSpeed变量的值，就可以改变小蛇的速度。

1. **难点：**设置各个不同道具的出现时机与用途。

**解决过程：**利用if语句判断条件来投递道具和实现相应的功能。当scores达到500时，投递生命药水道具，当生命药水道具rect与蛇头重合时，scores+100，该生命药水消失，蛇身增长一块；当1000 <= scores <= 1500 或者 2500 <= scores <= 3000，投递加速剂道具，当加速剂道具rect与蛇头重合时，scores+500，snakeSpeed+5，该加速剂道具消失，蛇身增长一块；当蛇身大于10时，投递毒药道具，当毒药道具rect与蛇头重合，scores-100，该毒药道具消失，调用addnode2（）函数，在小蛇速度相反方向增加蛇块，使小蛇撞到自身而死亡，从而达到小蛇吃到毒药道具而中毒身亡的药效。

1. **难点：**合理设置小蛇的速度：小蛇的运动速度随着小蛇的生长而增加，但如何设置合理的速度直接影响到游戏的体验感。

**解决过程：**应该设置一个极限速度，否则速度过快，游戏难度系数则过高。当蛇身长度小于10，snakeSpeed=5+len（snake.body）/2，否则snakeSpeed=16。

# 代码及运行结果

1. 代码：

import pygame

import sys

import random

# 全局定义

SCREEN\_X = 800

SCREEN\_Y = 800

# 蛇类

# 点以25为单位

class Snake(object):

# 初始化各种需要的属性 [开始时默认向右/身体块x5]

def \_\_init\_\_(self):

self.dirction = pygame.K\_RIGHT

self.body = []

for x in range(5):

self.addnode()

# 无论何时 都在前端增加蛇块

def addnode(self):

left, top = (0, 0)

if self.body:

left, top = (self.body[0].left, self.body[0].top)

node = pygame.Rect(left, top, 25, 25)

if self.dirction == pygame.K\_LEFT:

node.left -= 25

elif self.dirction == pygame.K\_RIGHT:

node.left += 25

elif self.dirction == pygame.K\_UP:

node.top -= 25

elif self.dirction == pygame.K\_DOWN:

node.top += 25

self.body.insert(0, node)

# 在速度相反方向增加蛇块，使小蛇撞到自身而死亡

def addnode2(self):

left, top = (0, 0)

if self.body:

left, top = (self.body[0].left, self.body[0].top)

node = pygame.Rect(left, top, 25, 25)

if self.dirction == pygame.K\_LEFT:

node.left += 25

elif self.dirction == pygame.K\_RIGHT:

node.left -= 25

elif self.dirction == pygame.K\_UP:

node.top += 25

elif self.dirction == pygame.K\_DOWN:

node.top -= 25

self.body.insert(0, node)

# 删除最后一个块

def delnode(self):

self.body.pop()

# 死亡判断

def isdead(self):

# 撞墙

if self.body[0].x not in range(SCREEN\_X):

return True

if self.body[0].y not in range(SCREEN\_Y):

return True

# 撞自己

if self.body[0] in self.body[1:]:

return True

return False

# 移动！

def move(self):

self.addnode()

self.delnode()

# 改变方向 但是左右、上下不能被逆向改变

def changedirection(self, curkey):

LR = [pygame.K\_LEFT, pygame.K\_RIGHT]

UD = [pygame.K\_UP, pygame.K\_DOWN]

if curkey in LR + UD:

if (curkey in LR) and (self.dirction in LR):

return

if (curkey in UD) and (self.dirction in UD):

return

self.dirction = curkey

# 食物类

# 方法： 放置/移除

# 点以25为单位

class Food:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具

# 道具一： 生命药水类

# 道具用途：当玩家得分达到500时，出现该道具，当玩家吃到该道具时，得分加100

# 方法： 放置/移除

# 点以25为单位

class Life:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具二： 加速剂类

# 道具用途：得分加500，小蛇速度加快

# 方法： 放置/移除

# 点以25为单位

class Speed:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 道具三：毒药类

# 道具用途：得分减100，小蛇中毒死亡

# 方法：放置/移除

# 点以25为单位

class Poison:

def \_\_init\_\_(self):

self.rect = pygame.Rect(-25, 0, 25, 25)

def remove(self):

self.rect.x = -25

def set(self):

if self.rect.x == -25:

allpos = []

# 不靠墙太近 25 ~ SCREEN\_X-25 之间

for pos in range(25, SCREEN\_X - 25, 25):

allpos.append(pos)

self.rect.left = random.choice(allpos)

self.rect.top = random.choice(allpos)

print(self.rect)

# 显示文本

def show\_text(screen, pos, text, color, font\_bold=False, font\_size=60, font\_italic=False):

# 获取系统字体，并设置文字大小

cur\_font = pygame.font.SysFont("宋体", font\_size)

# 设置是否加粗属性

cur\_font.set\_bold(font\_bold)

# 设置是否斜体属性

cur\_font.set\_italic(font\_italic)

# 设置文字内容

text\_fmt = cur\_font.render(text, 1, color)

# 绘制文字

screen.blit(text\_fmt, pos)

def main():

pygame.init()

screen\_size = (SCREEN\_X, SCREEN\_Y)

screen = pygame.display.set\_mode(screen\_size)

pygame.display.set\_caption('Snake')

clock = pygame.time.Clock()

scores = 0

isdead = False

# 蛇/食物/道具

snake = Snake()

food = Food()

life = Life()

speed = Speed()

poison = Poison()

while True:

for event in pygame.event.get():

if event.type == pygame.QUIT:

sys.exit()

if event.type == pygame.KEYDOWN:

snake.changedirection(event.key)

# 死后按space重新

if event.key == pygame.K\_SPACE and isdead:

return main()

screen.fill((255, 255, 255))

# 画蛇身 / 每一步+1分

if not isdead:

scores += 1

snake.move()

for rect in snake.body:

pygame.draw.rect(screen, (20, 220, 39), rect, 0)

# 显示死亡文字

isdead = snake.isdead()

if isdead:

show\_text(screen, (200, 300), 'YOU DEAD!', (227, 29, 18), False, 100)

show\_text(screen, (250, 380), 'press space to try again...', (0, 0, 22), False, 30)

# 根据得分评定等级

if scores <= 500:

show\_text(screen, (250, 450), 'Your level is F', (161, 163, 166), False, 60)

elif 500 < scores <= 1000:

show\_text(screen, (250, 450), 'Your level is D', (0, 174, 157), False, 60)

elif 1000 < scores <= 1500:

show\_text(screen, (250, 450), 'Your level is C', (0, 154, 214), False, 60)

elif 1500 < scores <= 2000:

show\_text(screen, (250, 450), 'Your level is B', (253, 185, 51), False, 60)

else:

show\_text(screen, (250, 450), 'Your level is A', (247, 172, 188), False, 60)

# 食物处理 / 吃到+50分

# 当食物rect与蛇头重合,吃掉 -> Snake增加一个Node

if food.rect == snake.body[0]:

scores += 50

food.remove()

snake.addnode()

# 食物投递

food.set()

pygame.draw.rect(screen, (136, 0, 21), food.rect, 0)

# 当scores达到500时，出现生命药水道具

if scores >= 500:

# 生命药水处理 / 吃到scores+100分

# 当生命药水道具rect与蛇头重合,吃掉 -> Snake增加一个Node

if life.rect == snake.body[0]:

scores += 100

life.remove()

snake.addnode()

# 生命药水投递

life.set()

pygame.draw.rect(screen, (238, 153, 221), life.rect, 0)

# 当1000 <= scores <= 1500 或者 2500 <= scores <= 3000,出现加速剂道具:

if 1000 <= scores <= 1500 or 2500 <= scores <= 3000:

# 加速剂类处理 / 吃到scores+500分 / 速度加5

# 当加速剂类道具rect与蛇头重合，吃掉 -> Snake增加一个Node

if speed.rect == snake.body[0]:

scores += 500

snakeSpeed += 5

speed.remove()

snake.addnode()

# 加速剂类投递

speed.set()

pygame.draw.rect(screen, (142, 68, 173), speed.rect, 0)

# 当蛇身大于10时，出现毒药道具：

if len(snake.body) > 10:

# 毒药处理 / 吃到 scores-100分

# 当毒药道具rect与蛇头重合，吃掉中毒死亡

if poison.rect == snake.body[0]:

scores -= 100

poison.remove()

snake.addnode2()

# 毒药投递

poison.set()

pygame.draw.rect(screen, (239, 246, 79), poison.rect, 0)

# 显示分数文字

show\_text(screen, (50, 700), 'Scores: ' + str(scores), (223, 223, 223))

pygame.display.update()

# 控制小蛇的速度，蛇身长度越长速度越快

if len(snake.body) < 10:

snakeSpeed = 5 + len(snake.body) / 2

else:

snakeSpeed = 16

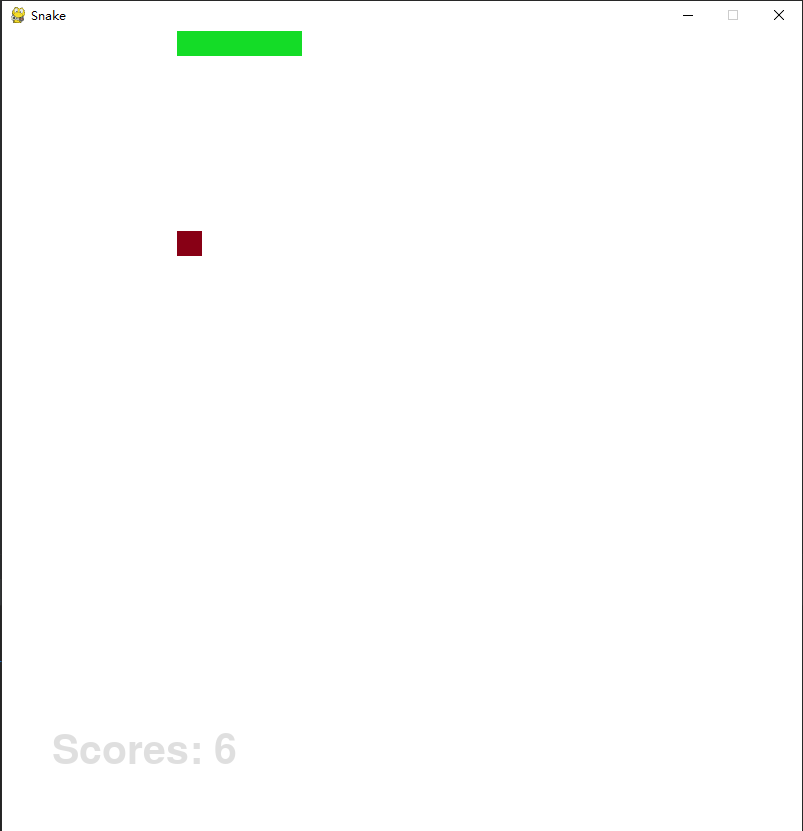
# clock.tick(10)

clock.tick(snakeSpeed)

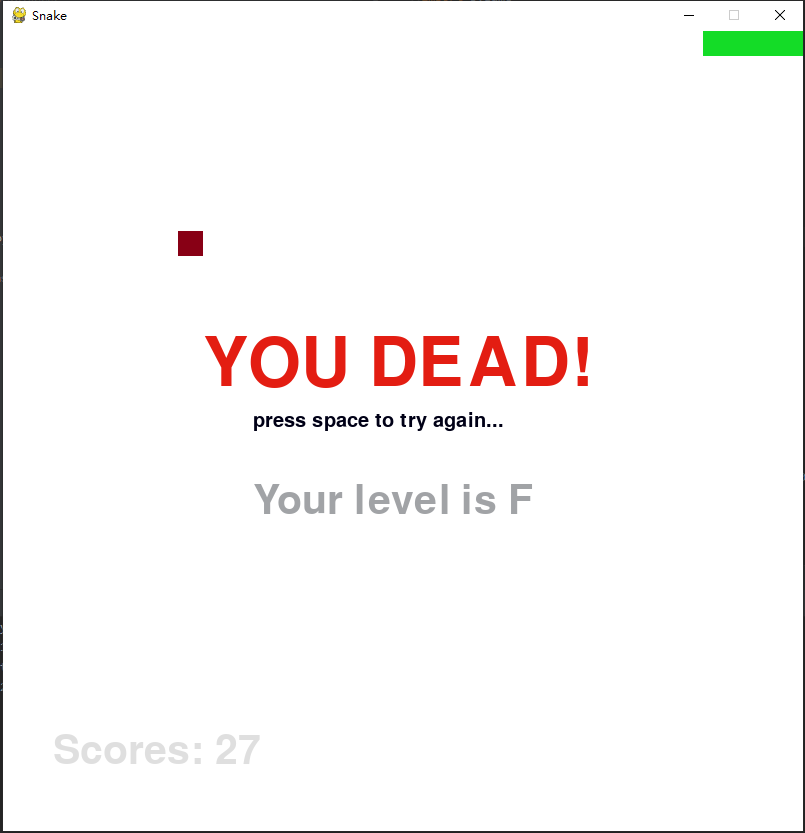
if \_\_name\_\_ == '\_\_main\_\_':

main()

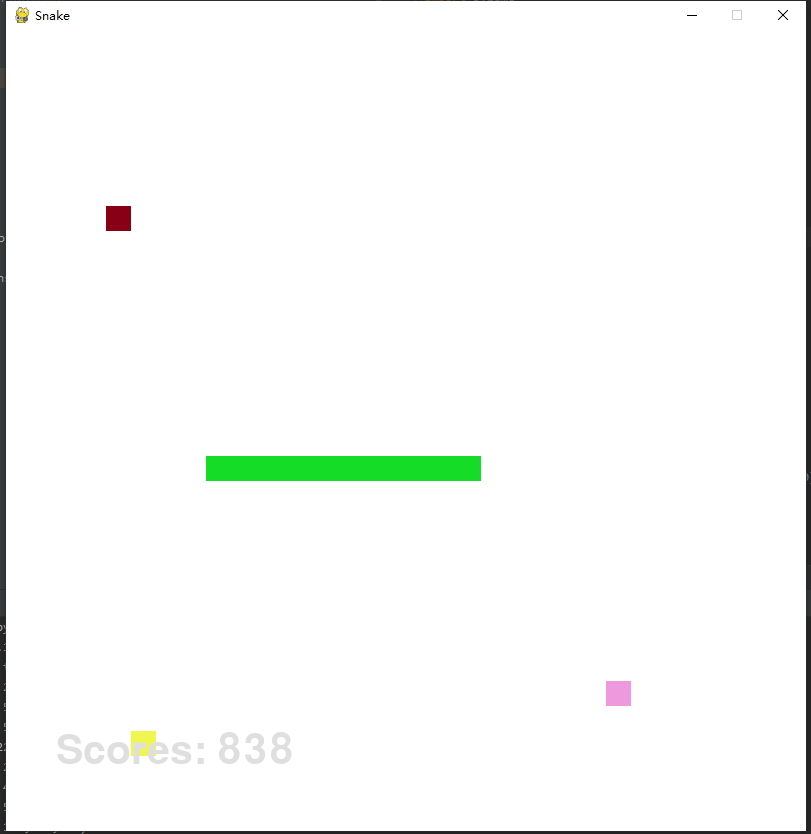
1. 运行结果：
   1. 初始界面



* 1. 结束界面



* 1. 出现道具



* 1. 不同等级评定

