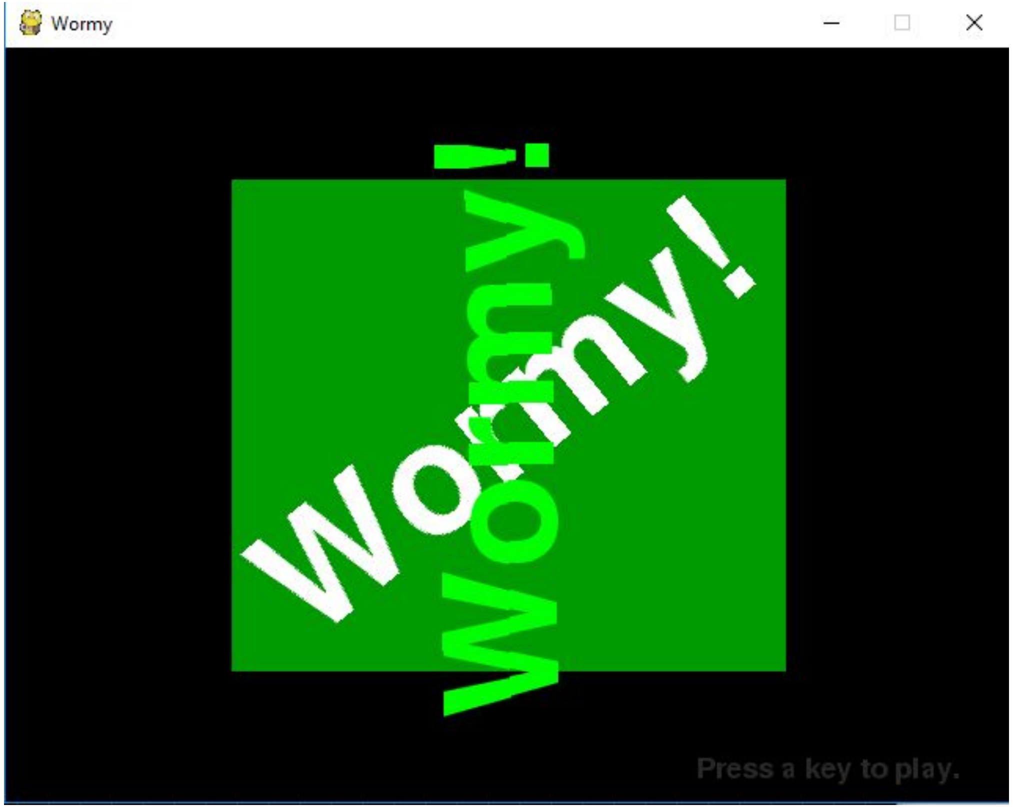
Wormy Game

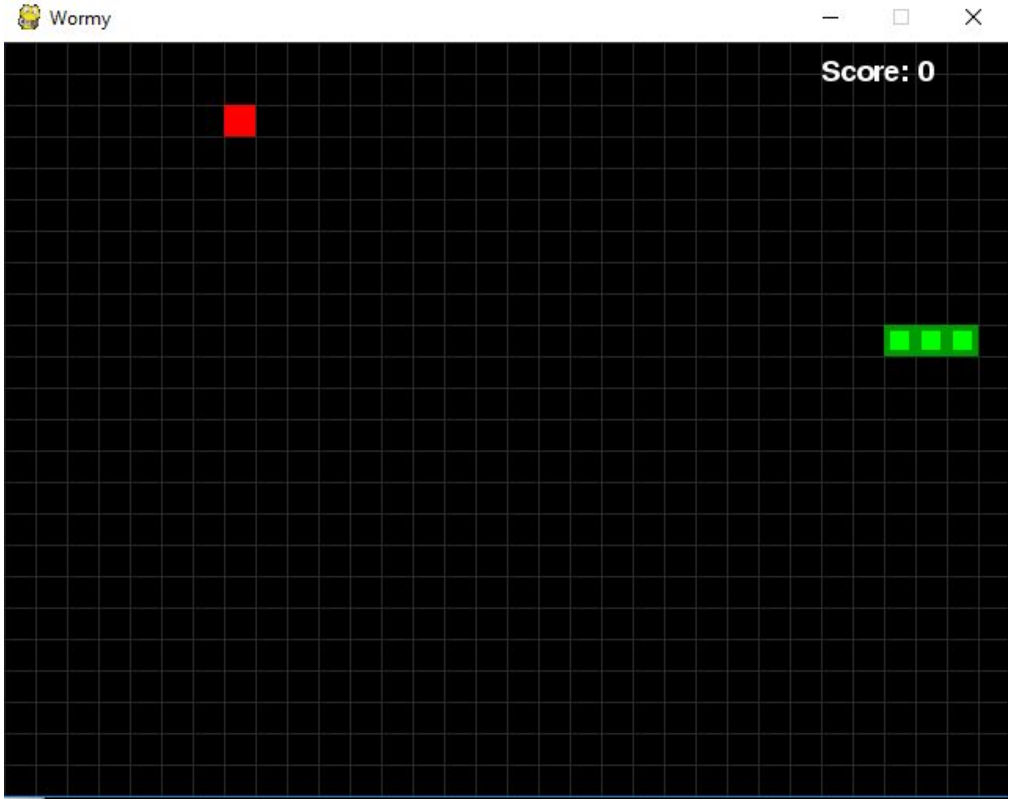
Game Begin



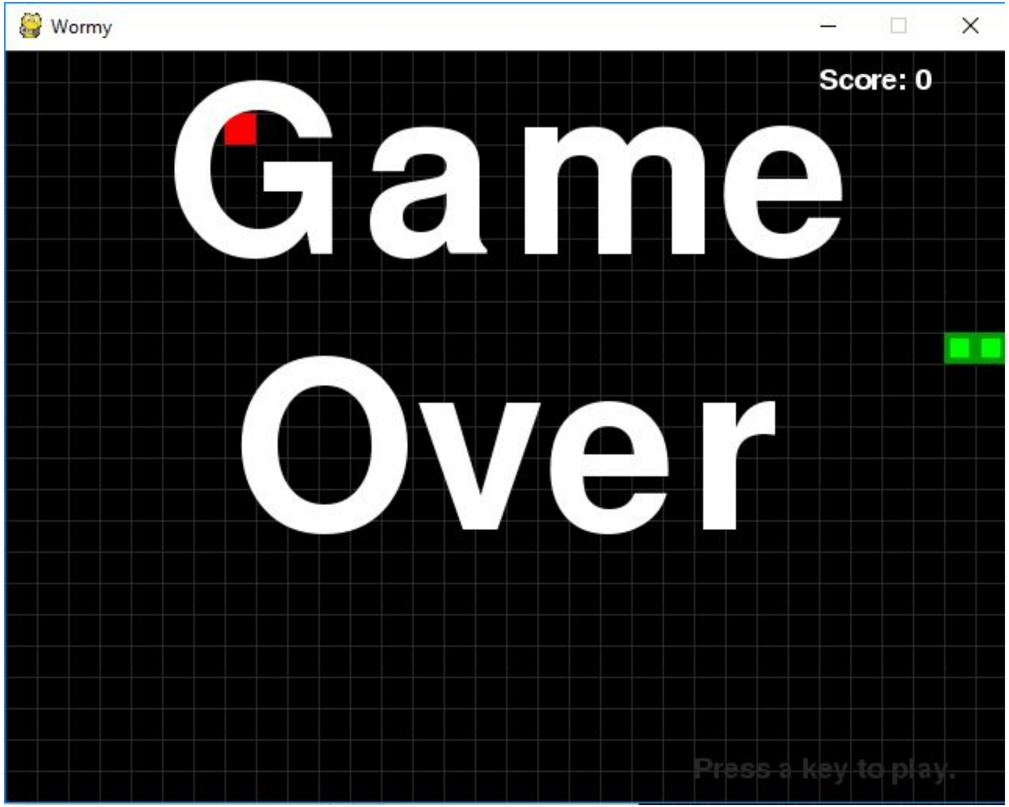
In the middle of game

Rules:

1. Eat the apple
2. Don’t hit the wall
3. Don’t hit worm body



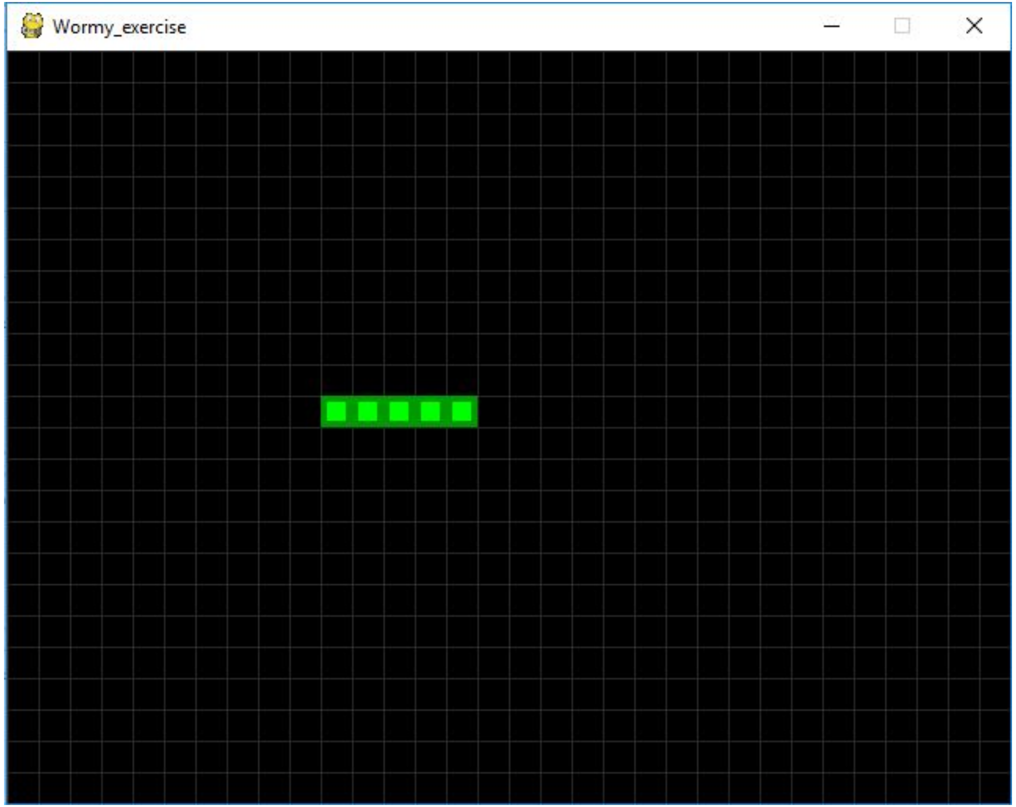
Game over



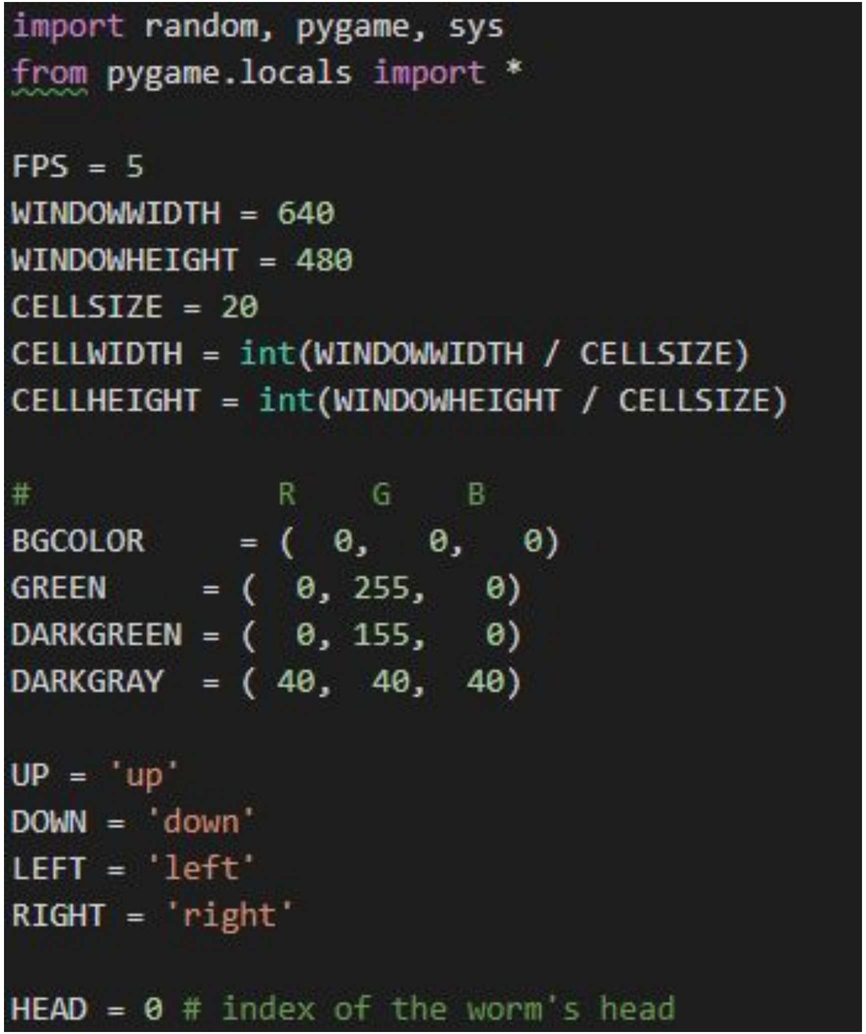
Game Time: 10 minutes

Winner: the highest score

Project 1: Training mode

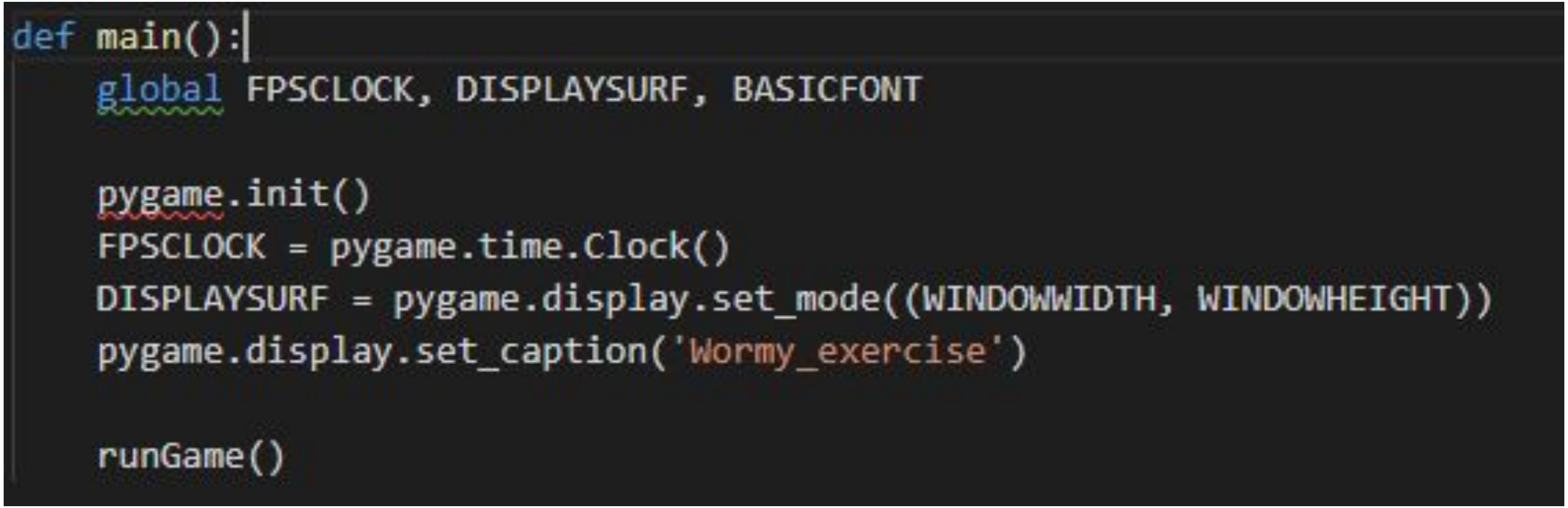


Code: setup



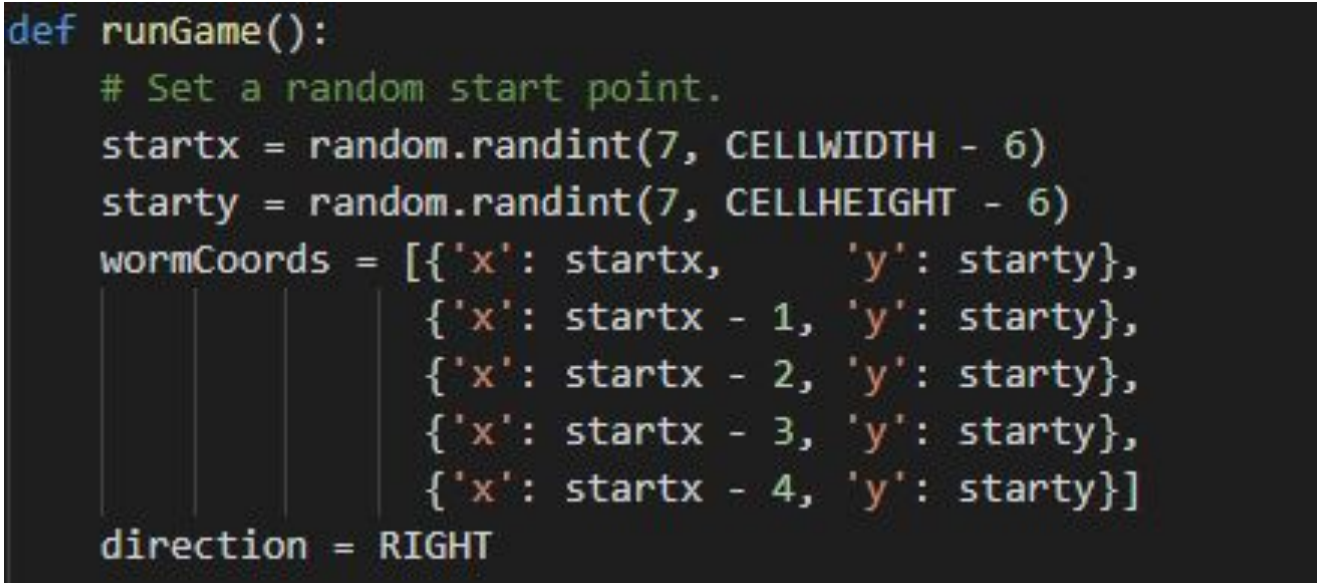
```
import random, pygame, sys
from pygame.locals import *

FPS = 5
WINDOWWIDTH = 640
WINDOWHEIGHT = 480
CELLSIZE = 20
CELLWIDTH = int(WINDOWWIDTH / CELLSIZE)
CELLHEIGHT = int(WINDOWHEIGHT / CELLSIZE)

#              R    G    B
BGCOLOR     = (  0,   0,   0)
GREEN     = (  0, 255,   0)
DARKGREEN = (  0, 155,   0)
DARKGRAY  = ( 40,  40,  40)

UP = 'up'
DOWN = 'down'
LEFT = 'left'
RIGHT = 'right'

HEAD = 0 # index of the worm's head
```

Main function



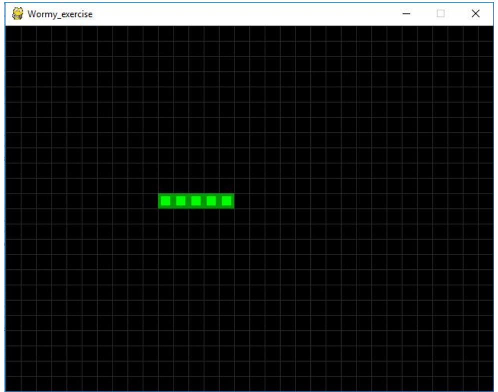
```
def main():
    global FPSCLOCK, DISPLAYSURF, BASICFONT

    pygame.init()
    FPSCLOCK = pygame.time.Clock()
    DISPLAYSURF = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT))
    pygame.display.set_caption('Wormy_exercise')

    runGame()
```

Data structure of worm

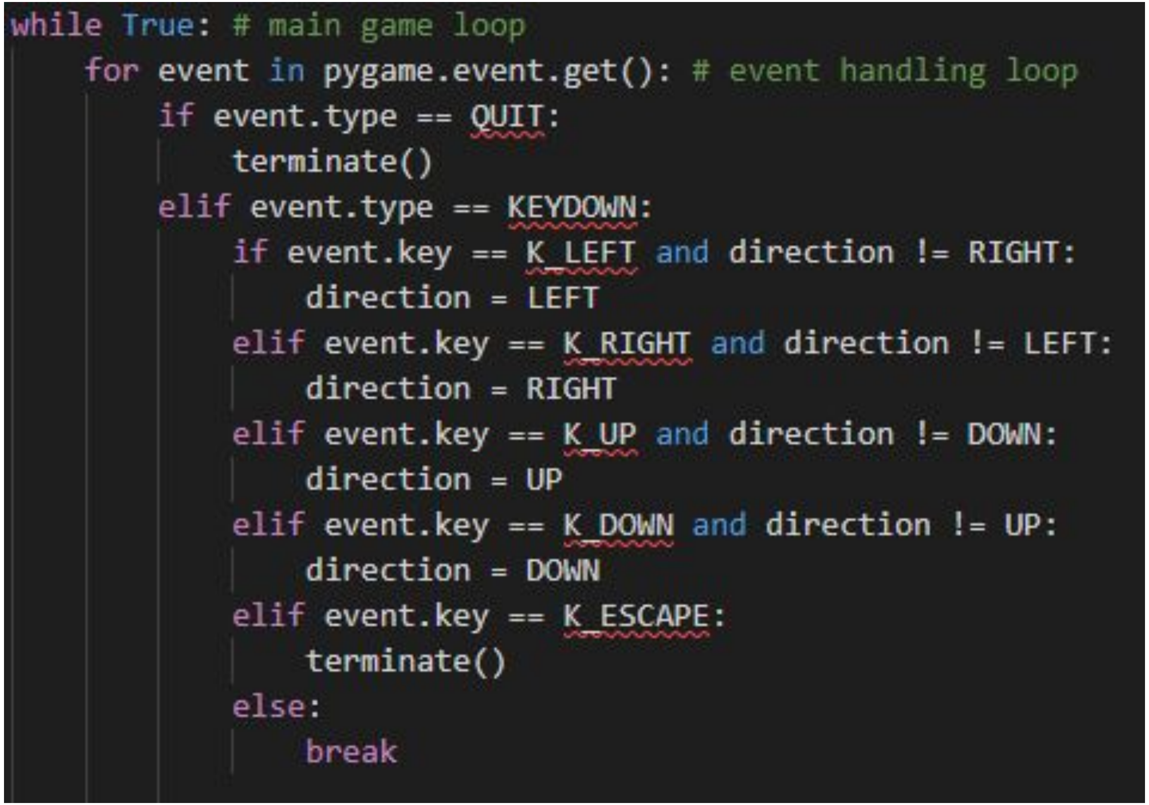


```
def runGame():
    # Set a random start point.
    startx = random.randint(7, CELLWIDTH - 6)
    starty = random.randint(7, CELLHEIGHT - 6)
    wormCoords = [{'x': startx,     'y': starty},
                  {'x': startx - 1, 'y': starty},
                  {'x': startx - 2, 'y': starty},
                  {'x': startx - 3, 'y': starty},
                  {'x': startx - 4, 'y': starty}]
    direction = RIGHT
```

4	3	2	1	0

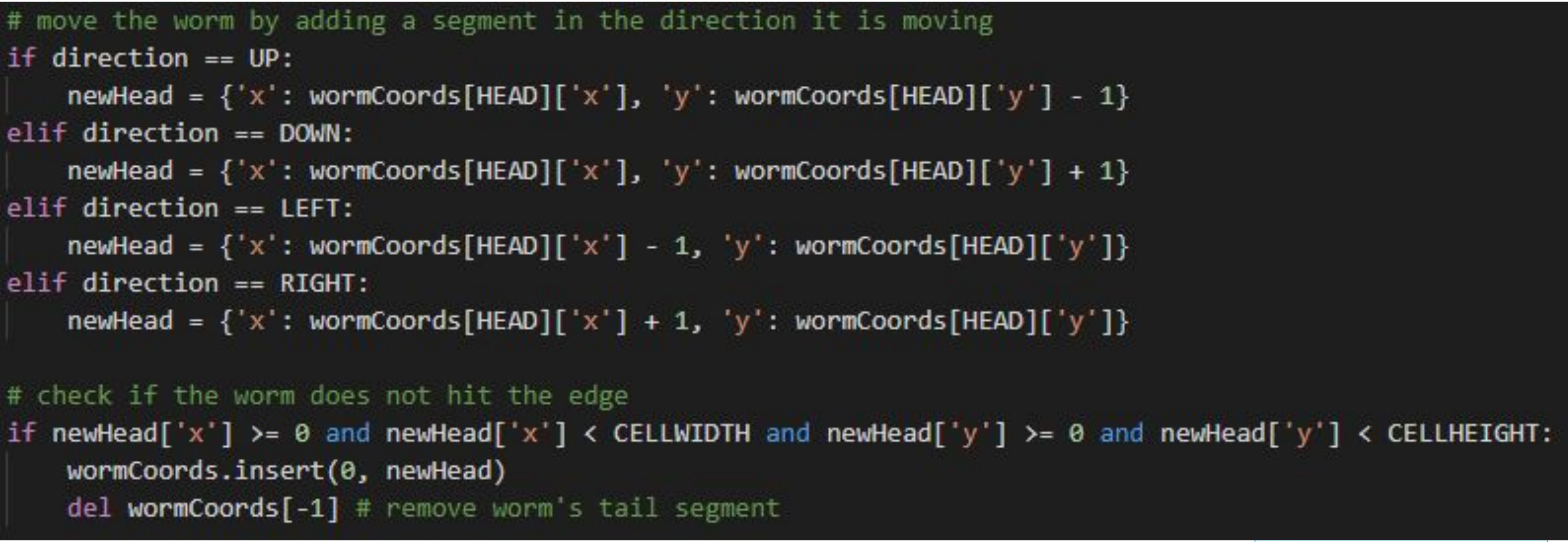


Main Loop: Event check



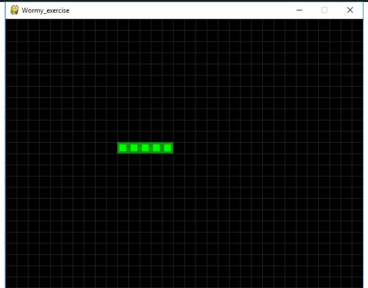
```
while True: # main game loop
    for event in pygame.event.get(): # event handling loop
        if event.type == QUIT:
            terminate()
        elif event.type == KEYDOWN:
            if event.key == K_LEFT and direction != RIGHT:
                direction = LEFT
            elif event.key == K_RIGHT and direction != LEFT:
                direction = RIGHT
            elif event.key == K_UP and direction != DOWN:
                direction = UP
            elif event.key == K_DOWN and direction != UP:
                direction = DOWN
            elif event.key == K_ESCAPE:
                terminate()
            else:
                break
```

Worm movement

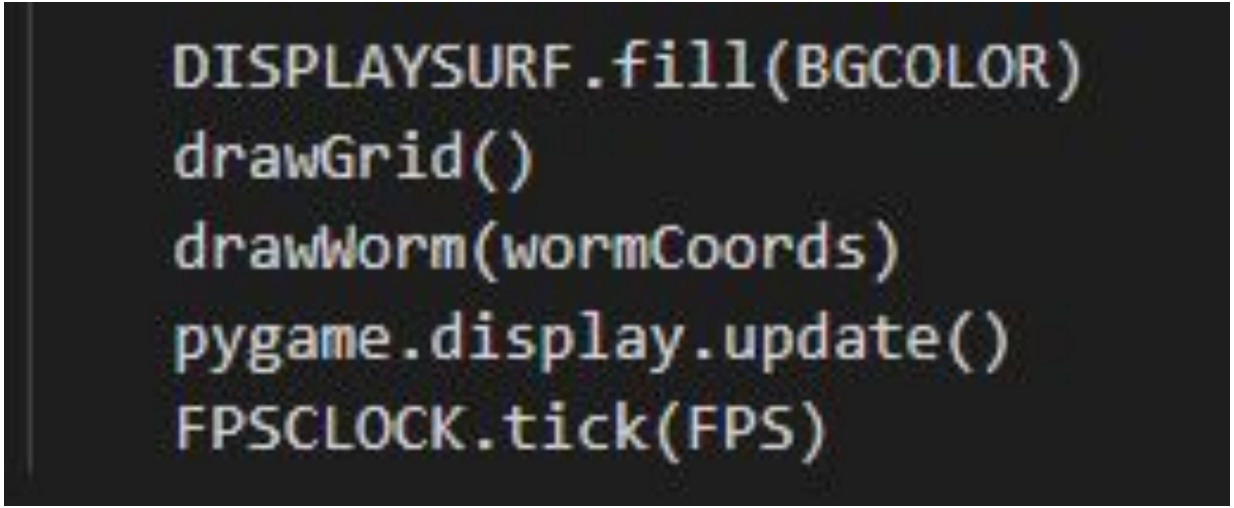


```python
# move the worm by adding a segment in the direction it is moving
if direction == UP:
    newHead = {'x': wormCoords[HEAD]['x'], 'y': wormCoords[HEAD]['y'] - 1}
elif direction == DOWN:
    newHead = {'x': wormCoords[HEAD]['x'], 'y': wormCoords[HEAD]['y'] + 1}
elif direction == LEFT:
    newHead = {'x': wormCoords[HEAD]['x'] - 1, 'y': wormCoords[HEAD]['y']}
elif direction == RIGHT:
    newHead = {'x': wormCoords[HEAD]['x'] + 1, 'y': wormCoords[HEAD]['y']}

# check if the worm does not hit the edge
if newHead['x'] >= 0 and newHead['x'] < CELLWIDTH and newHead['y'] >= 0 and newHead['y'] < CELLHEIGHT:
    wormCoords.insert(0, newHead)
    del wormCoords[-1] # remove worm's tail segment
```

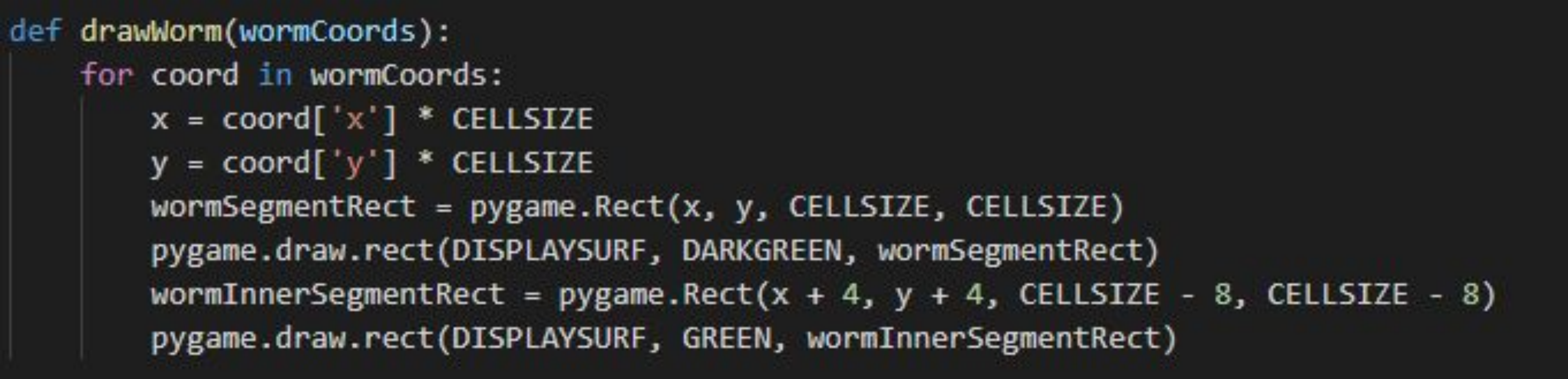
4	3	2	1	0	new



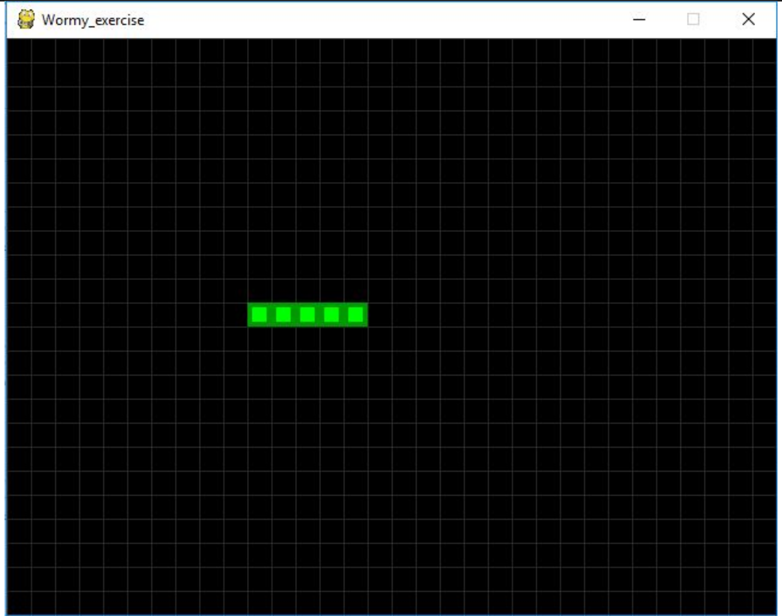
Main Loop: rendering

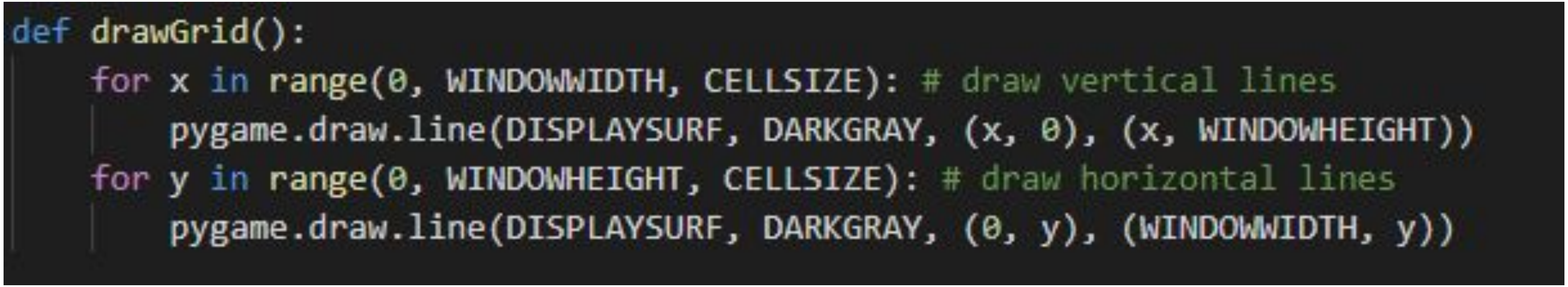


```
DISPLAYSURF.fill(BGCOLOR)
drawGrid()
drawWorm(wormCoords)
pygame.display.update()
FPSCLOCK.tick(FPS)
```

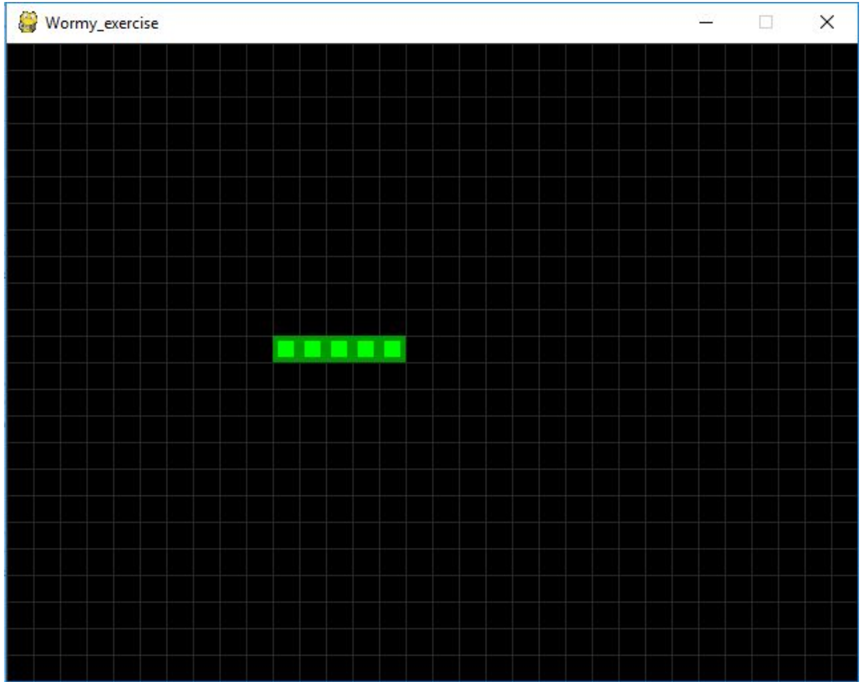


```python
def drawWorm(wormCoords):
    for coord in wormCoords:
        x = coord['x'] * CELLSIZE
        y = coord['y'] * CELLSIZE
        wormSegmentRect = pygame.Rect(x, y, CELLSIZE, CELLSIZE)
        pygame.draw.rect(DISPLAYSURF, DARKGREEN, wormSegmentRect)
        wormInnerSegmentRect = pygame.Rect(x + 4, y + 4, CELLSIZE - 8, CELLSIZE - 8)
        pygame.draw.rect(DISPLAYSURF, GREEN, wormInnerSegmentRect)
```





```python
def drawGrid():
    for x in range(0, WINDOWWIDTH, CELLSIZE): # draw vertical lines
        pygame.draw.line(DISPLAYSURF, DARKGRAY, (x, 0), (x, WINDOWHEIGHT))
    for y in range(0, WINDOWHEIGHT, CELLSIZE): # draw horizontal lines
        pygame.draw.line(DISPLAYSURF, DARKGRAY, (0, y), (WINDOWWIDTH, y))
```


Wormy_exercise