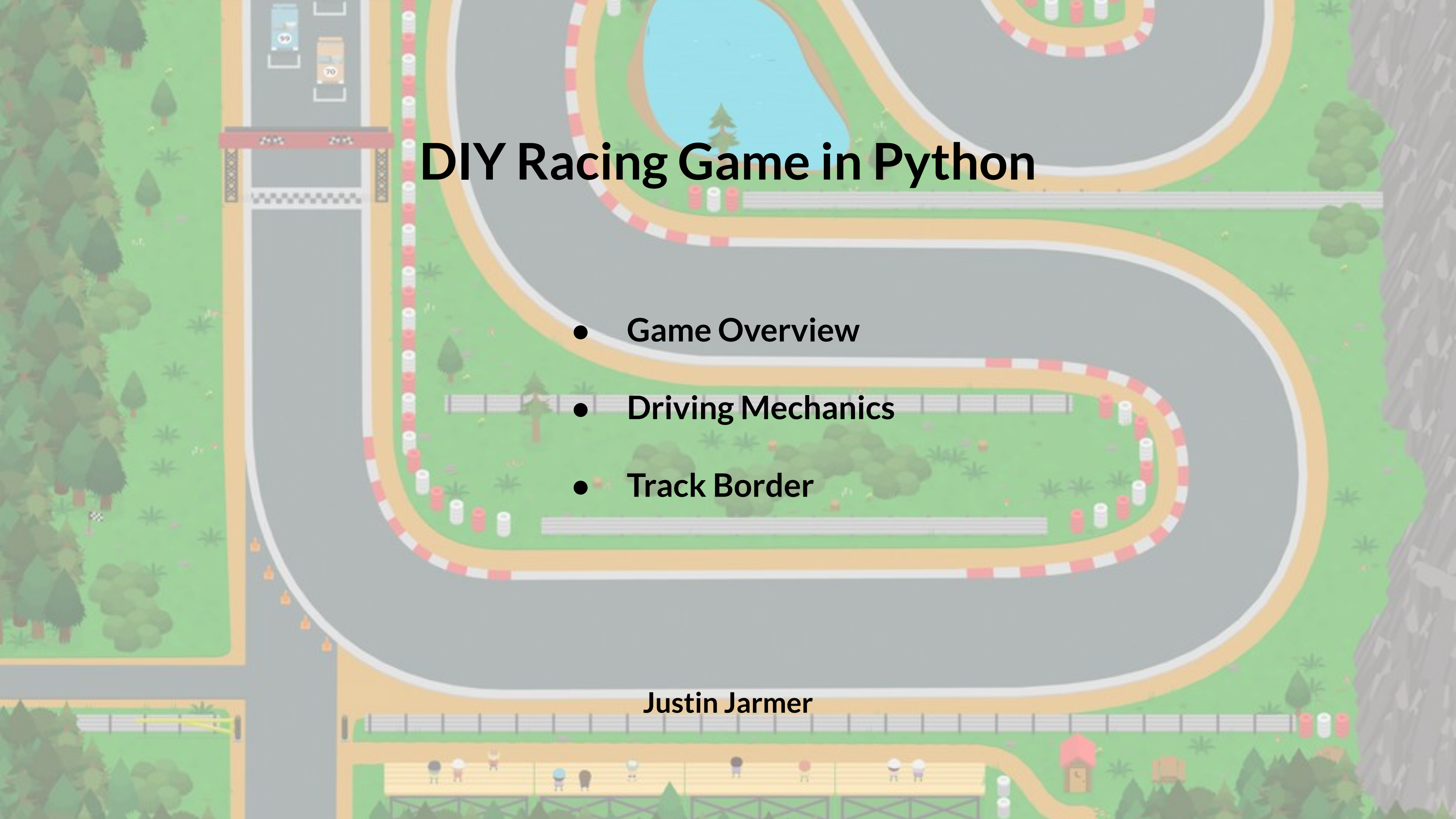
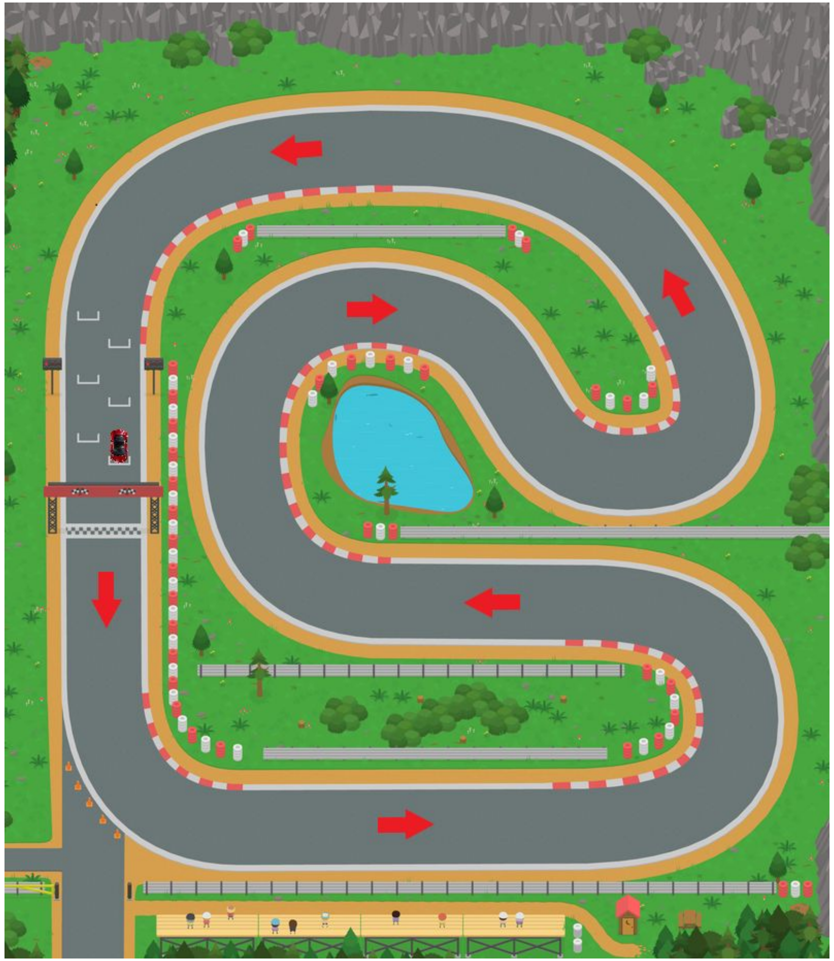


DIY Racing Game in Python

- Game Overview
- Driving Mechanics
- Track Border

Justin Jarmer



Overview:

Type:

- Single Player/Time Trial

Objective:

- Complete 5 laps as fast as possible

Controls:

Accelerate:	Up Arrow
Decclerate:	Space Bar
Turn CCW:	Left Arrow
Turn CW:	Right Arrow

Driving Mechanics

Original Vector/Magnitude:

- Direction of travel over last loop
- Distance traveled over last loop

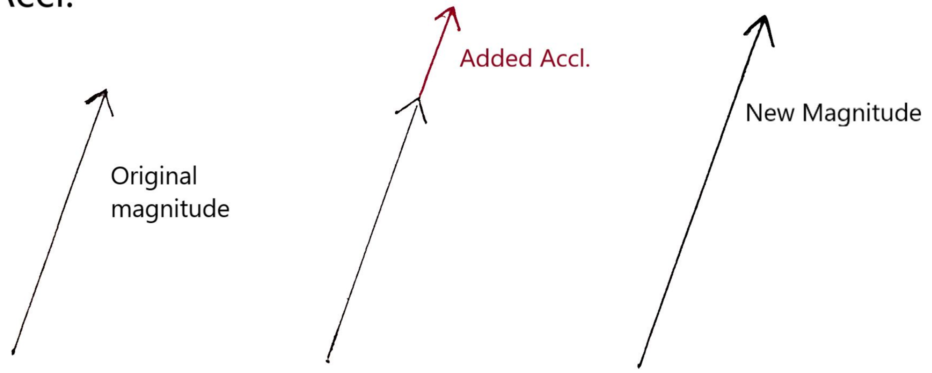
"Grip":

- Portion of Original Magnitude that changes with car's orientation
- Decreases with higher speeds

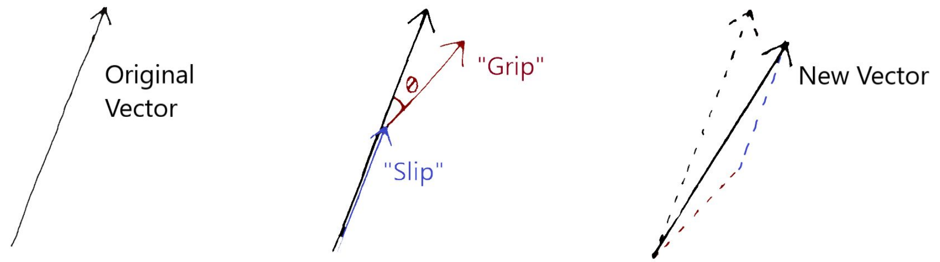
"Slip":

- Portion of Original Magnitude that remains the same
- Increases with higher speeds

+ or - Accl.



Change in Direction



Track Border

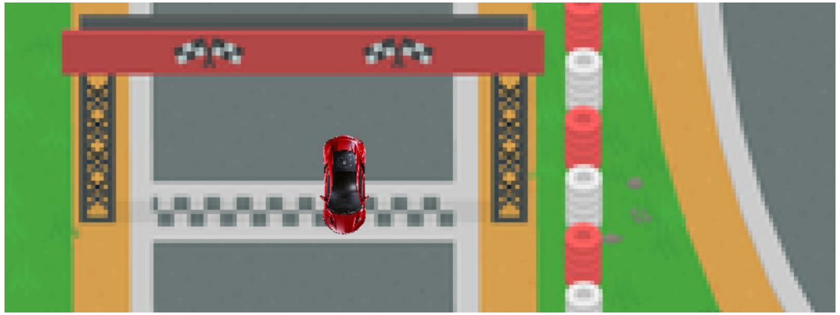
Collision Detection

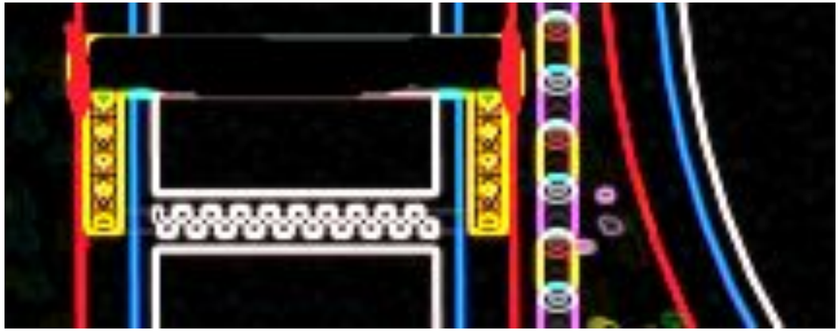
Available tools:

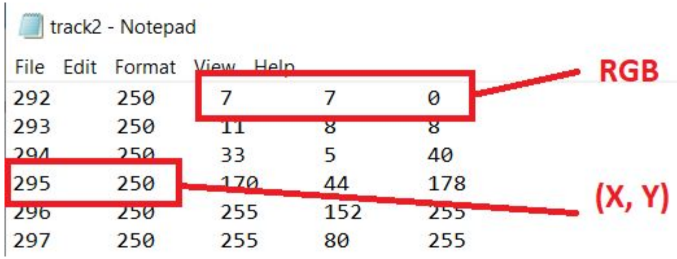
- Python Pygame: “Rectangle Collision”
- ImageJ: “Find Edges”
- ImageJ: “Save XY coordinates”

Summary:

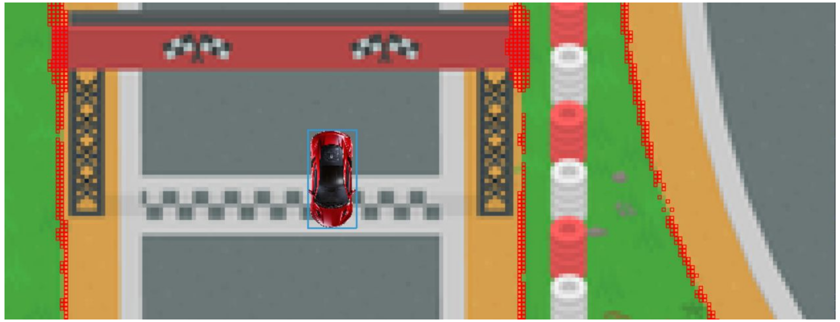
- Used Python to build rectangles for every red pixel in the image rendered in ImageJ

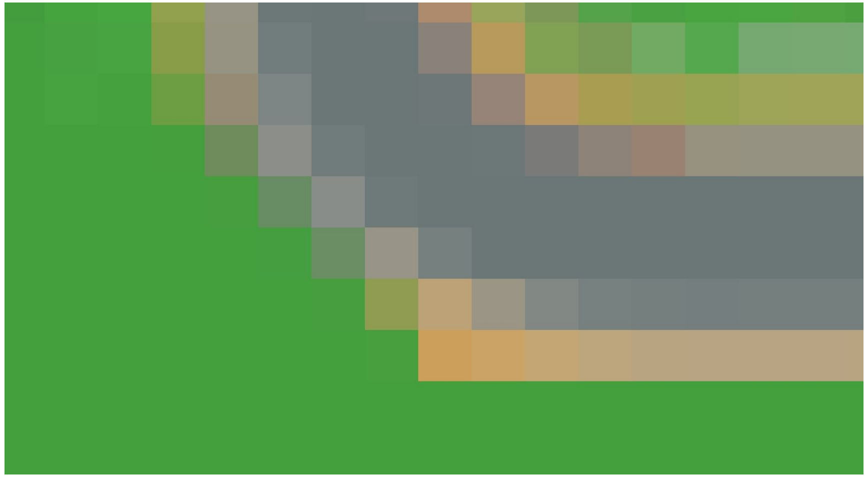


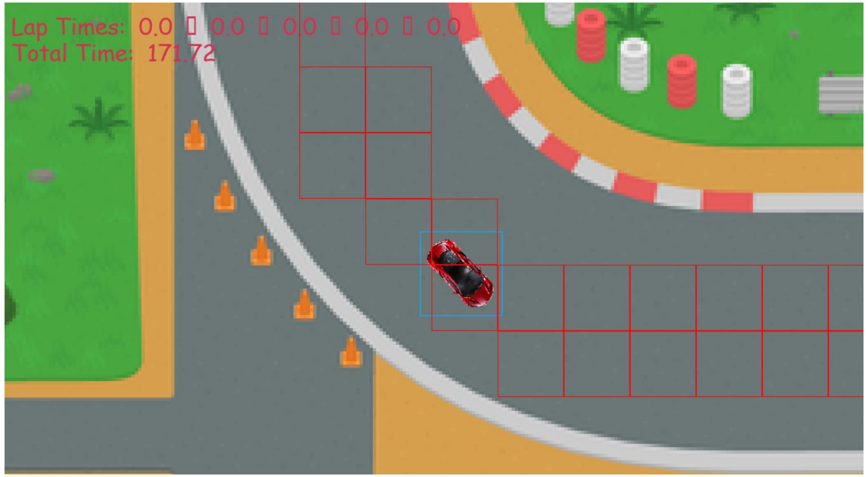




```
292    250    7      7      0
293    250    11     8      8
294    250    33     5      40
295    250    170    44     178
296    250    255    152    255
297    250    255    80     255
```







Track Border (cont.)

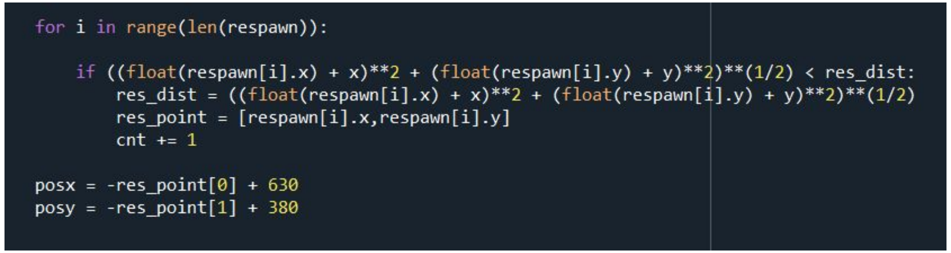
Respawn Points

General:

- Created like track border
- Uses distance formula to find closest

Issue:

- Orientation not reset upon respawn



```
for i in range(len(respawn)):

    if ((float(respawn[i].x) + x)**2 + (float(respawn[i].y) + y)**2)**(1/2) < res_dist:
        res_dist = ((float(respawn[i].x) + x)**2 + (float(respawn[i].y) + y)**2)**(1/2)
        res_point = [respawn[i].x,respawn[i].y]
        cnt += 1

posx = -res_point[0] + 630
posy = -res_point[1] + 380
```

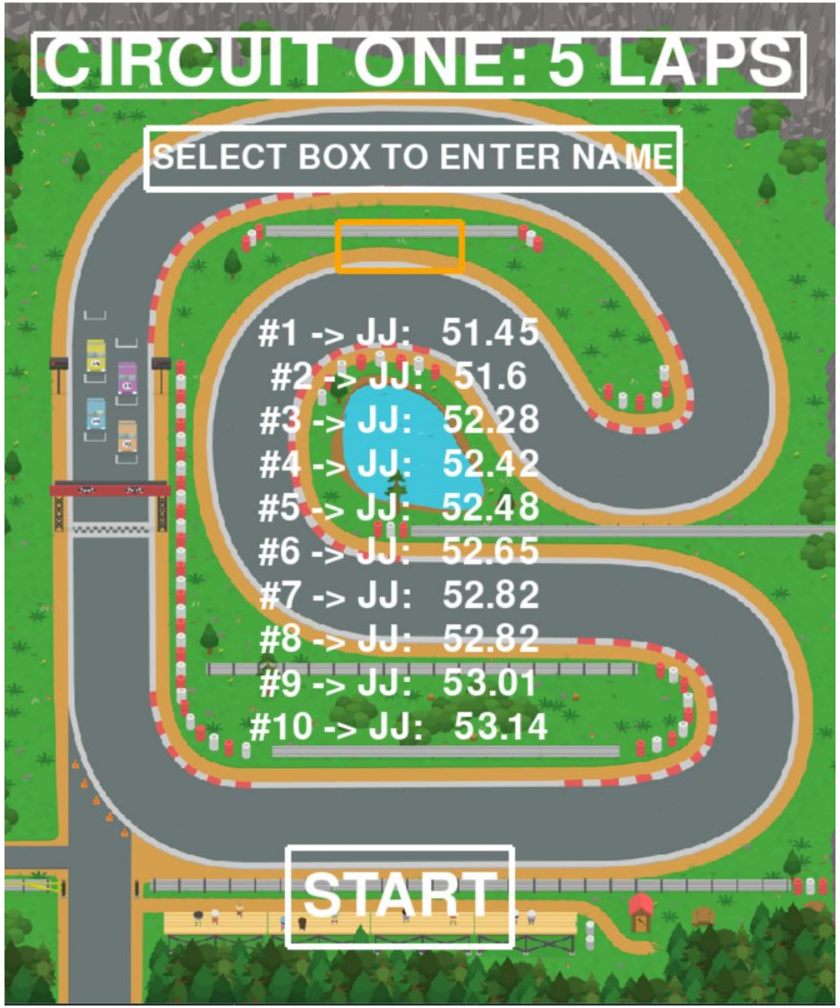
Conclusion

Summary:

- Developed Driving Mechanics aimed at increasing the skill ceiling
- Designed the track border to increase the consistency of the game

Acknowledgements:

- Track image: stock image from internet
- Youtube
- stack overflow



```
#1 -> JJ:  51.45
#2 -> JJ:  51.6
#3 -> JJ:  52.28
#4 -> JJ:  52.42
#5 -> JJ:  52.48
#6 -> JJ:  52.65
#7 -> JJ:  52.82
#8 -> JJ:  52.82
#9 -> JJ:  53.01
#10 -> JJ:  53.14
```