Project: Shall We Play A Game?

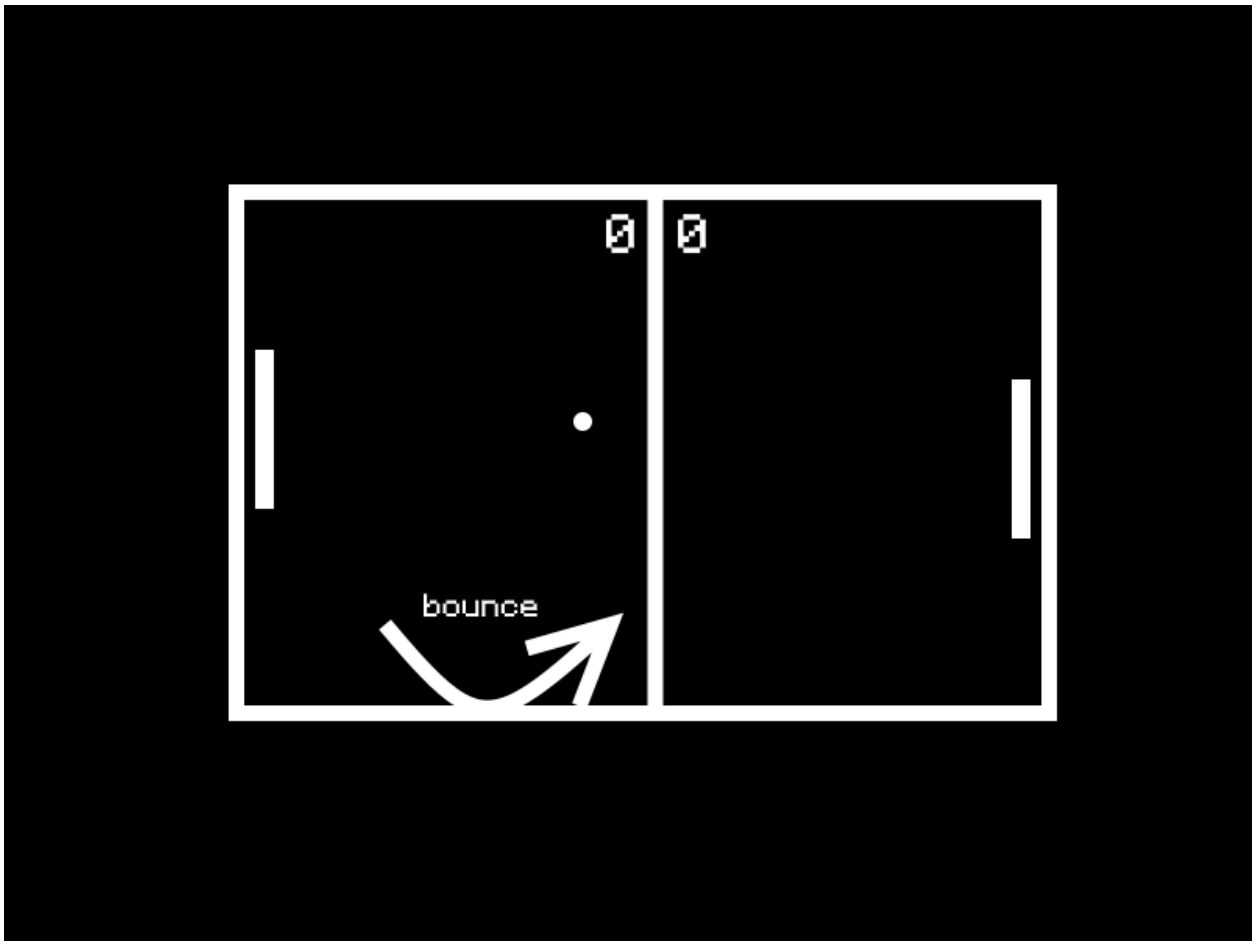
Computer Science Principles 2022-2023

by Leo Borrelli

Phase 1: Brainstorming

I have quite a bit of ideas for this project. A more favorable idea right now would be an idle simulator game which you may leave running while you are away from your computer. I've decided that I will be making a Pong game.

Prototype Drawing



Progress Plan

Day 1:

- Draw boundaries
- Draw necessary turtles

Progress

Day 1 (November 25th): I made the border. I had to take a bit to consider my design, so I changed the colors on the walls so that they ounce, and the backs are a different shade of white so I can use that to detect score.

Day 2 (November 28th): I get stuck on an issue where the paddles did not move. This was eventually fixed (after a couple of days) by listening for keypresses before the main game function.

Day 4 (November 30th): I got the ball in and it moves at different angles.

Day 5 (December 1st): Paddles now bounce at random angles. The game is finished.

Code Screenshots

ACTIVITY 125 > 125a.py > ...

```
106             time.sleep(1)
107             ballinplay = True
108             bounce = False
109             while ballinplay == True:
110                 pball.forward(3)
111                 cury = pball.ycor()
112                 if (cury <= -297):
113                     anglern = pball.heading()
114                     bruhangle = anglern
115                     pball.setheading(360-bruhangle)
116                 elif cury >= 297:
117                     anglern = pball.heading()
118                     bruhangle = anglern
119                     pball.setheading(360-bruhangle)
120                 if (pball.xcor() >= 397):
121                     p1score += 1
122                     p1scorewriter.clear()
123                     p1scorewriter.write(str(p1score), align="right", font=scorefont)
124                     ballinplay = False
125                     play_ball()
126                 elif (pball.xcor() <= -397):
127                     p2score += 1
128                     p2scorewriter.clear()
129                     p2scorewriter.write(str(p2score), align="left", font=scorefont)
130                     ballinplay = False
131                     play_ball()
132                 if (pball.distance(p1paddle) < 60):
133                     print(bounce)
134                     if bounce == False:
135                         bounce = True
136                         anglern = pball.heading()
137                         bruhangle = anglern
138                         # pball.setheading((360-bruhangle)+180)
139                         pball.setheading(random.randint(315, 360+45))
140                 elif (pball.distance(p2paddle) < 60):
141                     if bounce == False:
```

respond to events (bounce on coordinate)