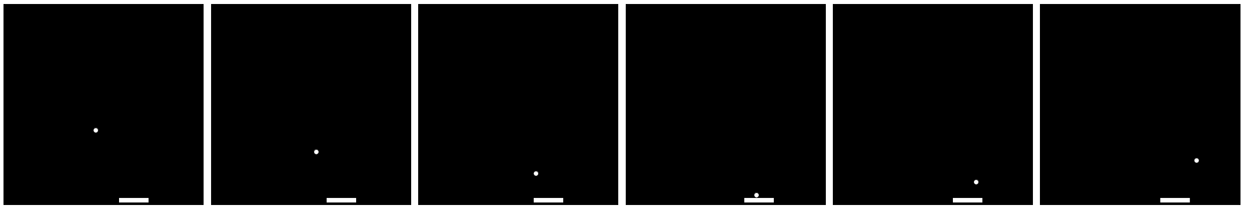
PAT 204/504: Creative Coding (Fall 2024)

Homework 2: Paddle Ball Game

Due at 11:59pm ET on September 13

1 Paddle Ball Game (5 points)

In this assignment, you will build a simple paddle ball game in Processing. The goal of this assignment is to get familiar with functions and if statements, the basic building blocks of a program. Below is a sequence of screenshots that showcase how the game works.

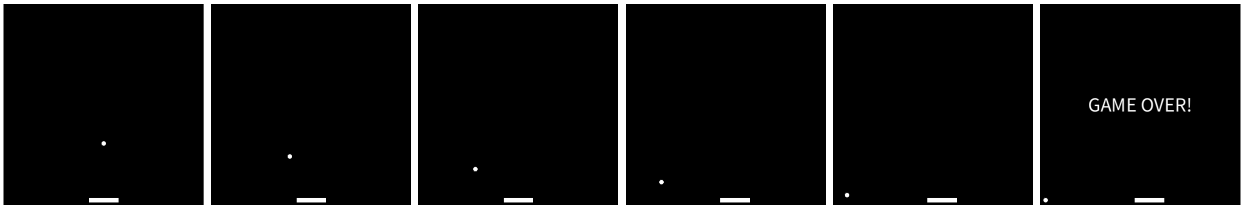


Specifically, this is how it works:

- When the game starts, the ball moves towards a random direction at a preset speed `ballSpeed` (defaults to 10).
- The player controls the paddle bar's horizontal movement using the mouse. Specifically, the paddle bar should be at the same x-position as the mouse.
- The ball bounces back when it hits the top/left/right wall.
- The ball bounces back if it hits the paddle bar at the bottom.
- The game is over if the player doesn't catch the ball with the paddle bar.
- The game *restarts* when the player clicks the mouse. The ball is reinitialized to move towards another random direction.

Below is a sequence of screenshots that showcase what happens when the player did not catch the ball. You can also find an animated example here.

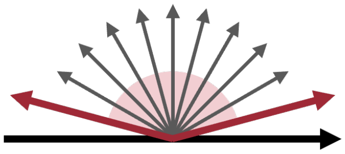
Please use the code template attached at the end. Specifically, you will need to implement the `init()` function and complete the `setup()` and `draw()` functions.



Please submit both your code and your final rendering as an animated GIF. You will receive zero credit if the code is missing. The template includes a boolean switch `saveFrames` that controls if the `saveFrame()` function at the end of the `draw()` function will be executed. When `saveFrames` is set to `true`, each frame generated by a call of the `draw()` function will be saved to the folder `frames/` in the project directory. You can then use the built-in movie maker ("Tools" → "Movie Maker") to create an animated GIF.

Hints

- You might want to limit the initial direction of the ball to a smaller range than $[0, \pi]$ so that it won't take too long for the ball to reach to the top and back. See the figure below.



2 Bonus: Sticky & Shrinking Paddle Ball Game (1 extra point)

Now, let's make this game more fun and challenging:

- Shrink the paddle bar size after each successful catch until the bar reaches a reasonable minimum width.
- Apply *stickiness* between the paddle and the ball. That is, a fixed ratio of the bar velocity will be added to the ball.

You can find an animated example here. **Please submit both your code and your final rendering as an animated GIF.** You will receive zero credit if the code is missing. Try your best to demonstrate the stickiness by moving the paddle when catching the ball.

Hints

- You can calculate the paddle velocity using `pmouseX` and `mouseX`, where `pmouseX` stores the value of `mouseX` in the previous frame.

3 Submission

- All assignments must be completed on your own. You are welcome to exchange ideas with your peers, but this should be in the form of concepts and discussion, not in the form of writing and code.
- Please provide proper citations/references for any external resources you use in your writing and code.
- Please submit your work to Gradescope.
- Late submissions will be deducted by 1 point per day.

```
boolean isGameOver = false;  // whether the game is over
boolean saveFrames = false;  // whether to save the frames
float barWidth = 60;  // width of the paddle bar
float barHeight = 10;  // height of the paddle bar
float ballSpeed = 10;  // speed of the ball
float ballWidth = 10;  // width of the ball
// DEFINE OTHER GLOBAL VARIABLES YOU NEED HERE

// Initialize the game
void init() {
  // YOUR CODE HERE
}

void setup() {
  // Create a 400x400 canvas
  size(400, 400);

  // Set the frame rate to 30 fps
  frameRate(30);

  // YOUR CODE HERE

  // Initialize the game
  init();
}

void draw() {
  // Check if the game is over
  if (isGameOver) {
    // Save the frame for making an animated GIF
    if (saveFrames) {
      saveFrame(”frames/###.png”);
    }

    // Return immediately if the game is over
    return;
  }

  // YOUR CODE HERE

  // Save the frame for making an animated GIF
  if (saveFrames) {
    saveFrame(”frames/###.png”);
  }
}

// Reset the game when the mouse is clicked
void mouseClicked() {
  init();
}
```