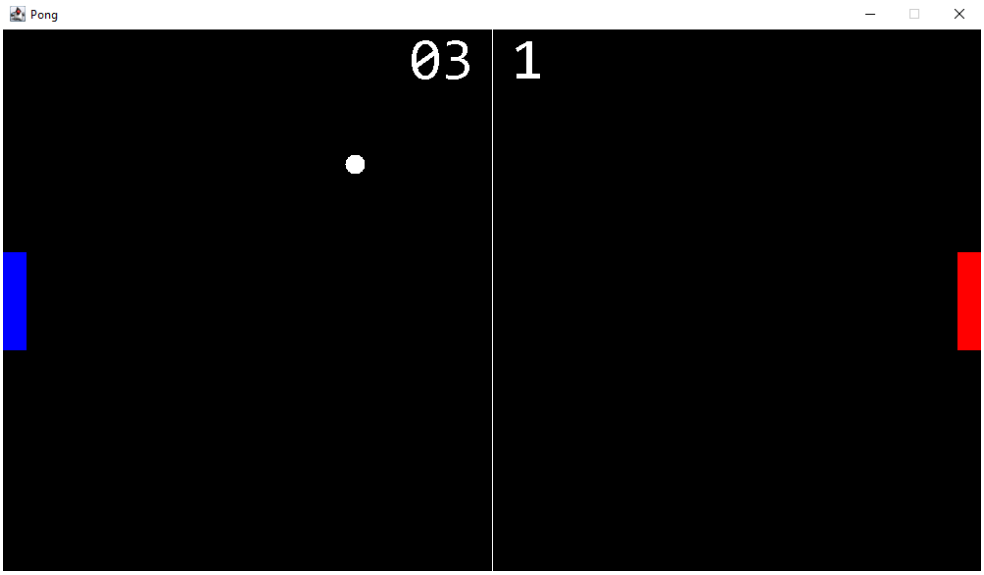
Project5 (200 points) **Due 4/30/2023**

Pong Hints



1.0 Pong Hints

The first task that you need to do is create your objects, and define some useful variables:

```
public class CanvasPanel_v1 extends JPanel implements MouseListener
{
    private final static int X_CORNER = 25;
    private final static int Y_CORNER = 25;
    private final static int CANVAS_WIDTH = 1000;
    private final static int CANVAS_HEIGHT = 555;
    private final static int BALL_DIAMETER = 25;

    private List<Shape2D> shapesList;
    private Rectangle2D   leftPaddle;
    private Rectangle2D   rightPaddle;
    private Rectangle2D   ball;

    private boolean       action;   // Simulate (on/off)
```

Next, you want to make sure you have a shape list for the polymorphic method `Draw`

```
public CanvasPanel_v1()
{
   shapesList = new ArrayList<>();
   action     = false;

   // paddles
   leftPaddle  = new Rectangle2D(5,   25, 250, 25, 100);
   rightPaddle = new Rectangle2D(5, 1000, 250, 25, 100);
   ball        = new Rectangle2D(5,  500, 250, 25,  25);
   ball.SetSpeed(3, 3);

   shapesList.add(leftPaddle);   // white rectangle, shape 0
   shapesList.add(rightPaddle);   // white rectangle, shape 1
   shapesList.add(ball);  // white ball rectangle, shape 3;
```

Render the shapes and scoreboard.

```
public void paintComponent(Graphics g)
{
..
..

   // Render all the shapes in the shapes list
   for (Shape2D shape : shapesList)
   {
      shape.Draw(g);
   }

  // Display frame number or scoreboard
  g.setColor(Color.white);
  g.setFont(new Font("Consolas", Font.PLAIN, 30));
  g.drawString(Integer.toString(frameNumber), 500, 70);
```

Finally, get things to move and react in the Simulate() method

```
public void Simulate()
{
    if (action)
    {
        ball.Animate();
    }

    rightPaddle.Animate();
    leftPaddle.Animate();

    // Collision Detection
    int ballX = ball.GetX();
    int ballY = ball.GetY();
    int ballW = ball.GetWidth();
    int ballH = ball.GetHeight();

    // Bounce ball off top and bottom of gameboard
    if ( (ballY > (CANVAS_HEIGHT)) || (ballY <= 0) )
    {
       ball.NegateSpeedY();
    }

    // Check for collision with right paddle
    int deltaX = Math.abs(ballX - paddX);
    int deltaY = ballY - paddY;

    boolean collide = (!(ballY > (paddY + paddH)));
    collide = collide && (!(paddY > (ballY + ballH)));

    if (collide)
    {
        if (deltaX <= BALL_DIAMETER)
        {
            ball.IcrSpeedX();
            ball.NegateSpeedX();
        }
    }

    if (ballX > CANVAS_WIDTH) // left player scores!!!!!!!!!!!
    {
        Reset();  // Set ball back to center and pause action
        // Update scoreboard
    }

    // Check for collision with left paddle
    ..
    ..
```

```
}
```

The `Reset` function is to reset the game state for the start of another volley. You might use the `RandomInteger` class to generate random ball trajectories. It might look like this:

```
public void Reset()
{
    ball.SetPos(500, 255);
    ball.SetSpeed(3, 2);
    action = false;
 }
```

You move the paddles by setting the speed with the callback methods for the keys. For example, you might use the up arrow and down arrow keys to move the right paddle:

```
public class myActionListener extends KeyAdapter
{
    public void keyPressed(KeyEvent e)
    {
        switch (e.getKeyCode())
        {
            case KeyEvent.VK_UP:
               rightPaddle.SetSpeed(0, -10);
            break;
            case KeyEvent.VK_DOWN:
               rightPaddle.SetSpeed(0, 10);
            break;
```

When you release the arrow buttons the paddle stops moving:

```
public void keyReleased(KeyEvent e)
{
    switch (e.getKeyCode())
    {
        case KeyEvent.VK_UP:
        case KeyEvent.VK_DOWN:
            rightPaddle.SetSpeed(0, 0);
        break;
        default:
            leftPaddle.SetSpeed(0, 0);
    }
}
```

I use the a-key and the s-key to toggle action to `true` and `false`. I use the `w` and s keys to move the left paddle. Once you have things working you do have the option to let the computer play against you autonomously. Maybe let the computer control the left paddle.