INTRO JAVASCRIPT

Alysson Diniz dos Santos

Codigo ate agora

```
function colorRect(topLeftX, topLeftY, boxWidth, boxHeight, fillColor) {
    canvasContext.fillStyle = fillColor;
    canvasContext.fillRect(topLeftX, topLeftY, boxWidth, boxHeight);
  }

  function colorCircle(centerX, centerY, radius, fillColor) {
    canvasContext.fillStyle = fillColor;
    canvasContext.beginPath();
    canvasContext.arc(centerX, centerY, radius, 0, Math.PI*2, true);
    canvasContext.fill();
  }
```

```javascript
function calculateMousePos(evt) {
    var rect = canvas.getBoundingClientRect();
    var root = document.documentElement;

    // account for the margins, canvas position on page, scroll amount, etc.
    var mouseX = evt.clientX - rect.left - root.scrollLeft;
    var mouseY = evt.clientY - rect.top - root.scrollTop;
    return {
      x: mouseX,
      y: mouseY
    };
}
```

```javascript
window.onload = function() {
    canvas = document.getElementById('gameCanvas');
    canvasContext = canvas.getContext('2d');

    var framesPerSecond = 30;
    setInterval(function() {
        move();
        draw();
     }, 1000/framesPerSecond);

    canvas.addEventListener('mousemove', function(evt) {
        var mousePos = calculateMousePos(evt);
        // minus half paddle height to center
        paddle1Y = mousePos.y - (PADDLE_HEIGHT/2);
      } );
   }
```

```
function move() {
    if(ballX < 0) { // if ball has moved beyond the left edge
        ballSpeedX *= -1; // reverse ball's horizontal direction
    }

    if(ballX > canvas.width) { // if ball has moved beyond the right edge
        ballSpeedX *= -1; // reverse ball's horizontal direction
    }

    if(ballY < 0) { // if ball has moved beyond the top edge
        ballSpeedY *= -1; // reverse ball's vertical direction
    }

    if(ballY > canvas.height) { // if ball has moved beyond the bottom edge
        ballSpeedY *= -1; // reverse ball's vertical direction
    }

    ballX += ballSpeedX; // move the ball based on its current horizontal speed
    ballY += ballSpeedY; // same as above, but for vertical
}
```

```
function drawEverything() {
    // clear the game view by filling it with black
    colorRect(0, 0, canvas.width, canvas.height, 'black');

    // draw a white rectangle to use as the left player's paddle
    colorRect(0, paddle1Y, 10, PADDLE_HEIGHT, 'white'); ////

    // draw the ball
    colorCircle(ballX, ballY, 10, 'white');
}
```

Criem uma funçao chamada ballReset, que retorna a bolinha para o centro da tela. Para teste, invoquem ballReset sempre que a bolinha atingir o lado esquerdo da tela.

Como detectar se a bola esta batendo no pauzinho?
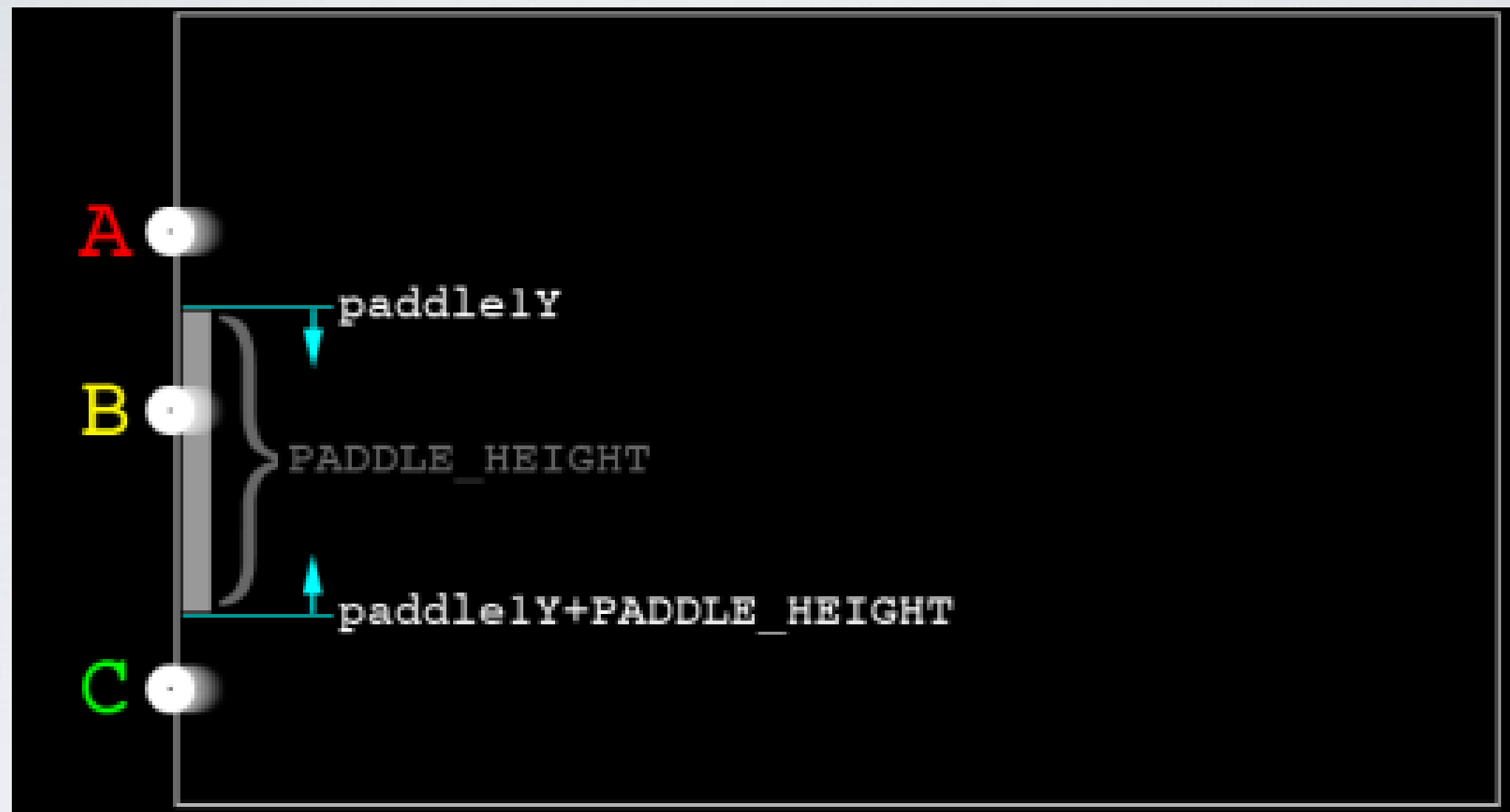
```
function move() {

    if (ballX <= 0) {
        if(ballY > paddle1Y && ballY < paddle1Y + PADDLE_HEIGHT)
            ballSpeedX = -ballSpeedX;
        else
            ballReset();
    }

    if (ballX >= canvas.width)
        ballSpeedX = -ballSpeedX;

    if (ballY >= canvas.height || ballY <= 0)
        ballSpeedY = -ballSpeedY;

    ballX = ballX + ballSpeedX;
    ballY = ballY + ballSpeedY;
}
```

Passem o controle do pauzinho da direita pro mouse, e testem se funciona

```
function move() {

    if (ballX <= 0) {
        if(ballY > paddle1Y && ballY < paddle1Y + PADDLE_HEIGHT)
            ballSpeedX = -ballSpeedX;
        else
            ballReset();
    }

    if (ballX >= canvas.width) {
        if(ballY > paddle2Y && ballY < paddle2Y + PADDLE_HEIGHT)
            ballSpeedX = -ballSpeedX;
        else
            ballReset();
    }

    if (ballY >= canvas.height || ballY <= 0)
        ballSpeedY = -ballSpeedY;

    ballX = ballX + ballSpeedX;
    ballY = ballY + ballSpeedY;

}
```

IA BÁSICA PARA O NPC

```
var npcStep = 6;

…

function npcMovement() {
    if (paddle2Y < ballY)
        paddle2Y += npcStep;
    else
        paddle2Y -= npcStep;
}

function move() {
    npcMovement();
    …
}
```
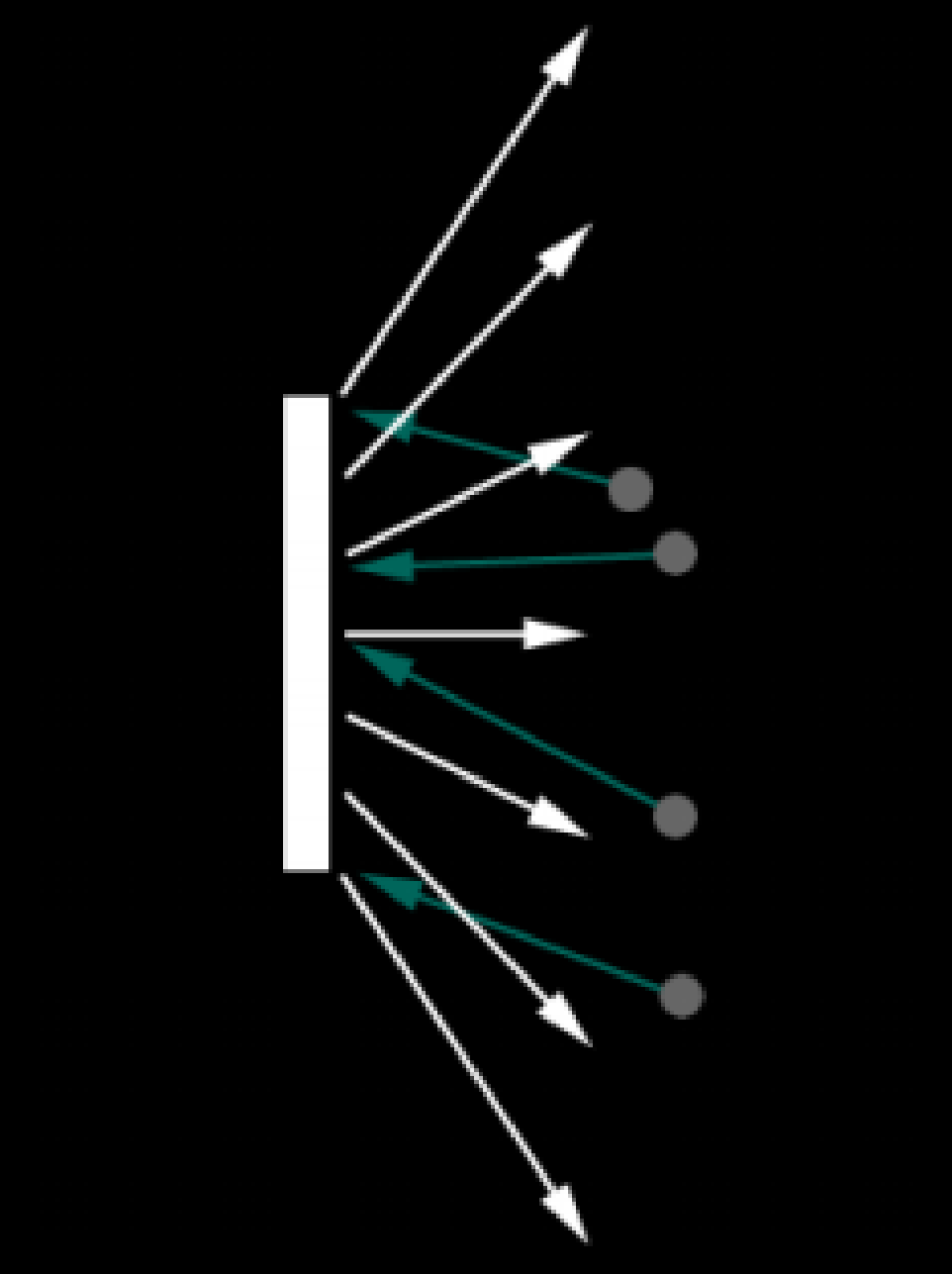
Como alinhar o pauzinho pelo centro?

```
var npcStep = 6;

…

function npcMovement() {
    var paddle2YCenter = paddle2Y + (PADDLE_HEIGHT/2);
    if (paddle2YCenter < ballY)
        paddle2Y += npcStep;
    else
        paddle2Y -= npcStep;
}

function move() {
    npcMovement();
    …
}
```

Alterando o movimento da bolinha

```
function move() {

    npcMovement();
    if (ballX <= 0) {
        if(ballY > paddle1Y && ballY < paddle1Y + PADDLE_HEIGHT) {
            ballSpeedX = -ballSpeedX;
            var deltaY = ballY – (paddle1Y+PADDLE_HEIGHT/2);
            ballSpeedY = deltaY*0.35;
        }
        else
            ballReset();
    }

    if (ballX >= canvas.width) {
        if(ballY > paddle2Y && ballY < paddle2Y + PADDLE_HEIGHT)
            ballSpeedX = -ballSpeedX;
        else
            ballReset();
    }
    …
}
```

COLOCANDO TEXTO NA TELA

```
function draw() {
    // black background
    colorRect(0,0,canvas.width,canvas.height,'black');
    // left player paddle
    colorRect(0,paddle1Y,10,100,'white');
    // ball
    colorCircle(ballX,ballY,10,'white');

    canvasContext.fillText("teste",100,100)
}
```

Exercicio: termine o jogo pong. Inclua uma linha vertical cetralizada e tracejada, score para o player o para o npc, e faça melhorias na IA do npc.