

Raquete do oponente

Unidade

Lógica de programação: desenvolvendo um jogo estilo Pong

Questão 1 – Colisão do oponente

Até o momento no jogo Pong, podemos visualizar o código abaixo dentro da função **draw**:

```
function draw() {  
  background(0);  
  mostraBolinha();  
  movimentaBolinha();  
  verificaColisaoBorda();  
  mostraRaquete(xRaquete, yRaquete);  
  movimentaMinhaRaquete();  
  verificaColisaoRaquete(xRaquete, yRaquete);  
  mostraRaquete(xRaqueteOponente, yRaqueteOponente);  
  movimentaRaqueteOponente();  
  verificaColisaoRaquete(xRaqueteOponente, yRaqueteOponente);  
}
```

Sabendo disso, analise as alternativas abaixo e marque aquela que mostra a função correta que está relacionada à colisão da raquete do oponente.

- ☐ a) **marcaPonto()**
- ☐ b) **verificaColisaoRaquete (xRaquete, yRaquete)**
- ☐ c) **verificaColisaoRaquete(xRaqueteOponente, yRaqueteOponente)**
- ☐ d) **movimentaRaqueteOponente()**

Alternativa A, incorreta. A função *marcaPonto()* é responsável por marcar os pontos no placar.

Alternativa B, incorreta. A função *verificaColisaoRaquete()* verifica a colisão da raquete do jogador em vez da raquete do oponente.

Alternativa C, correta. A função *verificaColisaoRaquete(xRaqueteOponente, yRaqueteOponente)* verifica se há colisão nos eixos X e Y da raquete do oponente.

Alternativa D, incorreta. A função *movimentaRaqueteOponente()* está relacionada com o movimento da raquete do oponente.

Questão 2 – Controle da raquete do oponente com a tecla W

Durante o desenvolvimento do jogo Pong, você programou a raquete do jogador para se movimentar utilizando as setas direcionais (para cima e para baixo). Para tornar o jogo multiplayer e movimentar a raquete do oponente, é necessário escolher outras teclas, que podem ser representadas por um código.

Ordene **todos os blocos** que movimentam a raquete do oponente para cima quando a tecla w (código 87) é pressionada.

`yRaqueteOponente -= 10; }`

`if`

`(keyIsDown(87)) {`

Escreva a sequência correta de blocos nas linhas a seguir:

Sequência correta: `if | (keyIsDown(87)){ | yRaqueteOponente -= 10; }`

Comentário: cada tecla é vinculada a um código, que pode ser identificado pela condicional e gerar a ação de movimento da raquete do oponente. A raquete só deve se mover quando a tecla w for pressionada, então a condição precisa estar correta dentro do bloco *if*.

Questão 3 – Invertendo a velocidade da bolinha

Para tornar o jogo mais interessante, é possível implementar pequenas modificações que alteram o nível de dificuldade. Pensando nisso, Joana quer aplicar as seguintes ideias:

- Toda vez que a bolinha colidir com uma das raquetes, a velocidade da bolinha no eixo X deve ser multiplicada por -1;
- A cor de fundo da tela deve ser alterada para vermelho.

Ordene **todos os blocos** que implementam as modificações de Joana.

`if (colidiu) {`

`background("red"); }`

`velocidadeXBolinha *= - 1;`

Escreva a sequência correta de blocos nas linhas a seguir:

Sequência correta: `if (colidiu) { | velocidadeXBolinha *= - 1; | background("red"); }`

Comentário: é possível incrementar o seu jogo aplicando diferentes modificações de cor, movimentação e velocidade dos elementos. Para alterar a cor de fundo, usamos a propriedade *background*.