

Exercício 11 – Chuckie Egg

Temática

- Variáveis acumuladoras
 - Estruturas de decisão
 - Estruturas de repetição
 - *Timers*
 - Sons
-

Na sua pasta de trabalho crie uma nova pasta com o nome “Exercicio_11”. Nesta pasta devem ser guardados os ficheiros desenvolvidos nesta aula.

Problema:

Pretende-se criar um jogo *online*, onde o jogador terá como objetivo apanhar todos os ovos que vão aparecendo. Deverá controlar os movimentos da personagem, recorrendo ao teclado, para que apanhe os ovos que se encontram espalhados pela área de jogo.

Pode consultar a versão resolvida em: <http://labmm.clients.ua.pt/LM3/LM3-p/ex11/>

Nível 1

Dentro da pasta “nivel1” e com base na página ‘index.html’ e no ficheiro ‘script.js’, desenvolva o código javascript necessário para implementar as alíneas seguintes.

Siga os seguintes passos para orientar uma possível solução para este jogo. Nesta primeira versão do jogo vamos construir uma versão simplificada, onde serão distribuídos aleatoriamente os ovos pela área de jogo e controlado o movimento do agricultor.

1. Desenvolver a função **carregaElementos** na qual deve:
 - a. Ser invocada quando todos os elementos da página html estiverem carregados;
 - b. Posicionar a 1ª plataforma afastada 150px do topo e a 2ª afasta 150px da 1ª;
 - c. Adicionar seis imagens ovo.png à div main;
 - d. Adicionar a imagem jogador_direita.png à div main.
2. Desenvolver a função **comecaJogar** na qual deve:
 - a. Ser invocada quando o utilizador clica no botão “jogar”;
 - b. Esconder a divAjuda;

- c. Posicionar aleatoriamente na área útil do *jogo* as imagens dos ovos e posicionar o agricultor na plataforma inferior e afastado 50px do limite esquerdo.
- 3. Desenvolver a função **processaTecla** na qual deve:
 - a. Ser invocada sempre que um evento de tecla pressionada é gerado¹;
 - b. Identificar qual a tecla pressionada². Se a tecla pressionada for:
 - w** – deve movimentar o agricultor para cima;
 - s** – deve movimentar o agricultor para baixo;
 - a** – deve movimentar o agricultor para a esquerda;
 - d** – deve movimentar o agricultor para a direita;
 - c. Garantir que o jogador apenas se movimenta horizontalmente nas plataformas e verticalmente nas escadas;
 - d. Modificar a imagem da personagem assumindo as imagens jogador_esquerda.png, jogador_direita.png e descer.png de acordo com a direção e o sentido do movimento que está a ser executado.

Nesta parte do exercício será desenvolvido o código necessário para que o agricultor possa apanhar os ovos.

- 4. Desenvolver a função **detectaColisao** na qual deve:
 - a. Verificar se agricultor toca nos ovos. Ao tocar num ovo, este deve desaparecer do ecrã e ser aumentado o contador de ovos `txt_ovos`;
 - b. Quando forem apanhados todos os ovos o jogo deve parar e ser apresentado o ecrã de ajuda.

Nesta parte do exercício deve adicionar sons³ ao movimento do agricultor:

- 5. Adicionar o som “tune.mp3” quando o jogo inicia;
- 6. Adicionar o som “updown.mp3” para a subida e descida de escadas;
- 7. Adicionar o som “egg.mp3” quando o agricultor apanha um ovo.
- 8. Adicionar o som “leftright.mp3” para os movimentos esquerda e direita.

Deve agora adicionar um tempo limite para o jogo, findo o qual aparece a `divAjuda` e o jogo termina.

Avançado

Altere agora a sua pasta de trabalho para a “nivel2”, aproveitando o código desenvolvido anteriormente (copie o ficheiro `script.js`) deve alterar o algoritmo forma a:

- 1. Garantir que a personagem se pode movimentar agora ao longo das três plataformas;
- 2. Aparecer um ovo por plataforma.

¹ <http://www.javascriptkit.com/javatutors/javascriptkey2.shtml>

² <http://www.javascripter.net/faq/keycodes.htm>

³ http://www.w3schools.com/tags/ref_av_dom.asp