

Exercício 7 – Toupeiras

Temática

- ☐ Variáveis acumuladoras
- ☐ Estruturas de decisão
- ☐ Timers
- ☐ Sons

Parte 1

Na sua pasta de trabalho crie uma nova pasta com o nome "Exercicio_7". Nesta pasta devem ser guardados os ficheiros desenvolvidos nesta aula.

Problema:

Pretende-se criar um jogo online, onde o jogador irá participar numa caça às toupeiras. As toupeiras são velozes e aparecem em locais inesperados. O objetivo do jogador é apanhar o máximo de toupeiras. De 1 em 1 segundo uma nova toupeira aparece num local aleatório do ecrã, até um máximo de 10 toupeiras. Durante esse tempo o jogador tenta clicar na toupeira, se o conseguir a toupeira é incrementado o número de toupeiras apanhadas. O jogo termina após terem aparecido 10 toupeiras.

Com base na página "index.html" e no ficheiro "script.js", desenvolva o código javascript necessário para implementar as alíneas seguintes. Antes de avançar para os pontos seguintes é importante que estude com atenção a estrutura do HTML, CSS e JS fornecido.

Versão resolvida: <http://labmm.clients.ua.pt/LM3/LM3-p/ex07/index.html>

Parte 1

Siga os seguintes passos para orientar uma possível solução para este jogo. Esta solução baseia-se no seguinte algoritmo simplificado.

1. Desenvolver a função **jogar** na qual deve:
 - a. A função **jogar** deve ser invocada quando o utilizador clica no botão "Jogar";
 - b. Esconder os elementos da interface que não devem estar visíveis na fase do jogo;
 - c. Inicializar os contadores de total de toupeiras saídas e toupeiras apanhadas;
 - d. Garantir que a função que vai desenvolver no ponto seguinte (**trocaPosicao**) é executada uma vez por segundo (recorra ao método `setInterval`¹).
2. Desenvolver a função **trocaPosicao** na qual deve:
 - a. Incrementar o contador de toupeiras saídas;
 - b. Garantir que a imagem da toupeira vai ser mostrada e corresponde a "mole.png";

¹ http://www.w3schools.com/jsref/met_win_setinterval.asp

- c. Mudar a toupeira para uma nova posição aleatória. A toupeira pode ser colocada em qualquer ponto do ecrã devendo utilizar as dimensões da janela do *browser* do jogador (utilize as propriedades **window.innerHeight** e **window.innerWidth**)². Tenha em conta as dimensões da toupeira para garantir que esta não é colocada fora da área visível da janela;
 - d. Verifique se já foi atingido o número máximo de toupeiras e, nesse caso, termine o jogo invocando a função **gameOver**.
3. Desenvolver a função **apanhou** na qual deve:
 - a. A função **apanhou** deve ser invocada quando o utilizador clica na imagem da "toupeira";
 - b. Incrementar e atualizar o contador de toupeiras apanhadas;
 - c. Alterar a imagem da toupeira para "mole-dead.png".
 4. Desenvolver a função **gameOver** na qual deve:
 - a. Terminar o mecanismo que temporiza³ o aparecimento das toupeiras;
 - b. Permitir que o jogador possa reiniciar o jogo.

Parte 2

5. Efetue as alterações necessárias no código desenvolvido, para que ao colocar o cursor em cima do botão "Jogar" a imagem seja trocada para "botaoJogarOver.png", ao retirar o cursor deve ser reposta a imagem inicial;
6. Melhore o jogo adicionando a componente sonora. Efetue as alterações necessárias no código desenvolvido para que quando a toupeira aparece seja reproduzido o som "toupeira.mp3" e quando a toupeira é apanhada deve ser reproduzido o som "boing.mp3".
7. Implemente um mecanismo que garante que uma toupeira só pode ser clicada uma vez por cada posição gerada (Sugestão: implemente um mecanismo de verificação com base numa variável booleana).

Avançado

Procure desenvolver uma solução que tenha por base a existência de 2 tempos de jogo distintos:

- as toupeiras continuam a ser geradas de 1 em 1 segundo;
- o tempo que cada uma está visível deve variar entre os 400ms e 800ms.

² http://www.w3schools.com/jsref/prop_win_innerheight.asp

³ http://www.w3schools.com/jsref/met_win_clearinterval.asp