

”

E-fólio A | Instruções para a realização do E-fólio

COMPUTAÇÃO GRÁFICA | 21020

Período de Realização

Decorre de 13 a 23 de novembro de 2020

Data de Limite de Entrega

23 de novembro de 2020, até às 23:55 de Portugal Continental

Trabalho a desenvolver:

Elabore uma página web em three.js e JavaScript que implemente os seguintes requisitos:

I- Implementação do algoritmo do ponto médio: Represente pontos bidimensionais por arrays de pares de números inteiros (portanto, em javascript, um ponto p será por exemplo $p=[2,3]$, em que a coordenada x do ponto é $p[0]$ ($p[0]=2$, neste caso) e a coordenada y é $p[1]$ ($p[1]=3$ neste caso).

Implemente uma função em javascript com as seguinte características (por favor siga exactamente a nomenclatura proposta!)

Nome da função: `lineMP`

Input: O input de `lineMP` é um par de pontos (arrays): $[x_1, y_1]$, e $[x_2, y_2]$. Portanto a chamada da função é da forma `lineMP([x1,y1],[x2,y2])`.

Output: O output é um array de pontos $n+1$ pontos (array de arrays 2D) $[[a_0, b_0], \dots, [a_n, b_n]]$ que deverá corresponder ao output do algoritmo do ponto médio para o segmento de recta que une os pontos do input (em particular temos que ter $[a_0, b_0]=[x_1, y_1]$ e $[a_n, b_n]=[x_2, y_2]$, ou seja, os pontos extremos coincidem com o input). Isto tem que funcionar para *todos* os casos, seja qual for a posição relativa dos pontos de input, seja qual for a inclinação da recta (incluindo verticais e horizontais).

A função `lineMP` deve ser guardada num ficheiro à parte, chamado **lineMP.js**, e que será chamado pelo seu `index.html`.

II- Interface Gráfico: Defina um plano no espaço 3D, que servirá de plano virtual de desenho 2D para desenhar os “pixels” gordos. Cada “pixel” desse plano será um cubo do three.js. Escolha um tamanho arbitrário para o tamanho do pixel gordo. Coloque-o numa variável chamada `fatPixelSize`.

Escolha um ponto como centro do referencia, e guarde as suas coordenadas num array chamado centerRef. Marque os dois eixos x e y usando rectas (finas) de three.js. Use rectas com outra cor para desenhar linhas “horizontais” e “verticais” a marcam a grelha dos pontos gordos (cada quadrado da quadricula deverá delimitar um pixel gordo). O plano do desenho será identificado no viewport por esta grelha de linhas finas. Inclua pelo menos uma grelha quadrada com 40 pixels de lado centrada no ponto centerRef.

3- Implemente um interface para o utilizador definir as coordenadas de dois pontos gordos, escrevendo as suas quatro coordenadas numa caixa de texto. Ao carregar num botão o programa deverá desenhar (com pixels gordos) o segmento de recta entre os dois pontos, que é definido pela função lineMP implementada acima. Note que as coordenadas de input são as do plano virtual de pixels gordos, e não necessariamente as coordenadas do espaço 3D do three.js.

4 – Implemente um botão que apaga as rectas já desenhadas.

5- Objectivo secundário: implemente um script que permita seleccionar os dois pontos extremos carregando com o mouse sobre o plano representado pelo three.js no viewport em vez de usar as caixas de texto.

6- Faça um viewport que mostre o plano de frente, de forma a que os pixels cúbicos pareçam quadrados (use uma camera de projecção ortográfica)

Os tamanhos dos pixels gordos, as linhas do referencial e o ponto de origem do plano devem ser escolhidos de forma a que os pixels sejam claramente identificáveis um a um, de forma a ler-se facilmente quais pixels estão acesos. O campo de visão deve permitir ver pelo menos um quadrado de 40 pixels gordos de lado, centrado em redor da origem.

7-Objectivo secundário: Faça também um segundo viewport que mostre uma vista inclinada sobre o plano, quer permita ver que os “pixels gordos” são cubos. Pode fazer como segundo viewport ou ter uma tecla que muda a perspectiva.

Siga à risca as nomenclaturas apresentadas acima para as funções e objectos. O seu algoritmo vai ser testado por um programa que assume esses nomes.

Deverá incluir todos os seus ficheiros num folder do ficheiro zip do seu trabalho. O seu programa deverá estar pronto para correr, sem dependências partidas. Inclua um ficheiro chamado index.html, que deverá correr tudo o resto. Todas as libraries necessárias deverão estar no sítio correcto da estrutura de ficheiros do zip. Poderá em alternativa carregar as libraries a partir de um repositório online, mas assegure-se que é algo estável, que ainda exista no momento da correcção da prova.

Não se esqueça de documentar o código em detalhe, usando comentários no javascript. Faça ainda um documento à parte onde explica as suas ideias e a sua implementação.

O seu trabalho será avaliado por vários factores:

- a implementação das features pedidas.
- a correcção, elegância, e eficiência do código e da documentação do mesmo.
- a correcção com que seguiu os requisitos de nomenclatura explicitados acima.

- requisito essencial: o programa tem que estar pronto a correr, sem alterações nem dependências quebradas.

Normas a respeitar

Todas as páginas da sua resolução devem ser numeradas. O texto deve ser claramente legível.

Programa os algoritmos e métodos usando JavaScript, Html, CSS e Three js e comente adequadamente o código fonte.

Junte no relatório explicativo de todas as decisões tomadas (se não entregar, haverá um desconto de 1.0 valor na nota final deste e-fólio). Este relatório constituirá uma memória descritiva do trabalho realizado, devendo ser sucinto (máximo de 2 páginas A4, fonte 10, espaçamento simples) e complementar os comentários inseridos no código fonte;

Faça entrega de todos os ficheiros comprimindo-os num único ficheiro zip, adotando o seguinte formato: <nome do aluno><numero de estudante>efolioA.zip.

Deve carregar o referido ficheiro para a plataforma no dispositivo E-fólio A até à data e hora limite de entrega. Evite a entrega próximo da hora limite para se precaver contra eventuais problemas.

Indique se detectar qualquer problema com o enunciado. Como o trabalho é individual, utilize o meu email em vez dos fóruns para qualquer dúvida cujo esclarecimento possa dar pistas impróprias aos seus colegas (na dúvida, assumo que é o caso): antonio.araujo@uab.pt

O ficheiro a enviar não deve exceder 8 MB.

Votos de bom trabalho!