

”

E-fólio A | Folha de resolução para E-fólio



UNIDADE CURRICULAR: Computação Gráfica

CÓDIGO: 21020

DOCENTE: António Araújo / Pedro Pestana

A preencher pelo estudante

NOME: Gonçalo Caraça

N.º DE ESTUDANTE: 2000130

CURSO: Engenharia Informática

DATA DE ENTREGA: 21/11/2022

TRABALHO:

1ª Parte

Dei início ao trabalho construindo o algoritmo do ponto médio com base na teoria que nos foi facultada na UC.

Comecei por criar um array vazio onde posteriormente irei armazenar os pontos, calcular os Deltas tanto para x como para y, iniciar as variáveis de decisão e calcular a diferença entre deltas e armazenar numa variável.

Depois com um Ciclo While em que o meu "pointsArray" faz push dos pontos, começo por verificar se o ponto atual é igual ao final, faço um break porque não há iterações para fazer.

De seguida defino a minha "var" decisão que é 2 vezes o auxiliar e inicio as iterações com "if's" até chegar as coordenados do ponto final e devolvo o meu Array com os pontos.

2ª Parte

Na segunda parte do trabalho, após definir todas as características necessárias para um layout 3D, camera, luz, renderer e controls, construi uma função (drawsquare) para definir as características de cada quadrado a apresentar na minha tela.

De seguida contrui a função (drawgrid) que recebe os parâmetros x, y, e cor para poder definir que cor irá receber cada quadrado, com dois ciclos for faço o controlo em x e y se são pares ou ímpares e chamo a função (drawsquare) para desenhar um quadrado e atribuir-lhe a respetiva cor de forma alternada para ter o efeito desejado.

De seguida criei uma função chamada "drawline" que serve única e exclusivamente para desenhar uma linha na cena, recebendo dois conjuntos de pontos e a cor desejada, função essa que utilizei na função "drawAxis" para desenhar os eixos do x e y com as cores pretendidas.

No passo seguinte usando o raycaster do three.js como solicitado, criei 3 variáveis para calcular os pontos das coordenadas onde o rato se encontra e os objetos de interseção, posteriormente com um if criei a condição necessária para apenas alterar as coordenadas quando se passasse com o rato para outro quadrado.

De seguida foi criar a função “onKeyPress” que armazena as coordenadas atuais ao pressionar a tecla “x” ou “X” desenhando um quadrado vermelho aproveitando a função (“drawsquare”) que já foi criada para desenhar o grid, com um if garanto que terá que ser acionado a tecla “x” duas vezes e então chamo a função “drawline” também já criada para desenhar os eixos, passo-lhe os dois conjuntos de pontos armazenados e a cor preta como solicitado para desenhar um traço entre estes pontos.

De seguida chamo a função “drawPoints” que por sua vez, chama a função lineMp, passando-lhe os dois conjuntos de pontos, finalizo a função “onKeyPress” limpo o array “xPoints” para se poder repetir o processo as vezes desejadas.

Uma nota para a minha função “drawPoints” acima mencionada, que com um “foreach”, para cada elemento do “pointsArray” chama a função “drawBox” para desenhar uma pequena caixa amarela transparente de ¼ de altura pelo grid.

Por fim com um if fui controlar se o evento acionado seria pressionar a tecla “Backspace”, se sim remove tudo da “scene” e chama novamente as funções “drawGrid” “drawAxis” e adiciona a luz para reiniciar tudo.

Anexos:

Programa.

Relatório.

Demo.

Fim