

”

UNIDADE CURRICULAR: Computação Gráfica

CÓDIGO: 21020

DOCENTE: António Araújo

A preencher pelo estudante

NOME: Hernâni Filipe Resendes Coelho

N.º DE ESTUDANTE: 1800045

CURSO: Engenharia Informática

DATA DE ENTREGA: 28/11/2021

TRABALHO / RESOLUÇÃO:

Este trabalho tem como objetivo apresentar uma página web onde é possível visualizar graficamente a aplicação do algoritmo do ponto médio para desenhar segmentos de reta. Para o seu desenvolvimento foi utilizado as linguagens de programação HTML e JavaScript, e as bibliotecas three e orbitcontrols.

A organização da estrutura de ficheiros da página está organizada de acordo com as indicações, pelo que resulta da seguinte forma:

- Diretório base
 - Index.html
 - lineMP.mjs
 - readMe.pdf
 - src
 - js
 - init.mjs
 - main.mjs

Organização do código:

O código está organizado de forma modular, de forma a otimizar a legibilidade, evitando que referências a linguagens diferentes constem do mesmo ficheiro, assim todas estas referências são invocadas de forma externa.

O Ficheiro index.html consiste num ficheiro simples que tem apenas a finalidade de fazer a importação do módulo inicial “init.mjs”. Assim que a janela do browser abrir, este módulo invoca a função “init” para iniciar a aplicação da página. Todas as funções da aplicação estão definidas no módulo “main.mjs”.

A função “init” chama as funções para criar a cena, criar o plano de 21 por 21 quadrados com cor alternada, criar as linhas dos eixos positivos x e y e posicionar a camara na cena. O utilizador tem a liberdade de utilizar o rato para deslocar o plano, rodar verticalmente e fazer zoom.

A aplicação recorre ao uso de handlers de eventos para tratar determinados eventos que são despoletados. Assim, o evento do movimento do rato é detetado e tratado pela função “onMouseMove”, que deteta e regista as coordenadas normalizadas. Uma vez que exista apenas movimento do rato sobre o tabuleiro, a função “obtainCoordinates” é invocada para detetar as interseções que existam entre o rato e os cubos que formam o tabuleiro, invocando depois a função “showCoordinates” para fazer logs das coordenadas para a consola.

Por outro lado, existe outro handler para tratar dos eventos de teclas pressionadas, e assim que uma tecla é pressionada é invocado a função “keyboardPress” que, caso a seja a tecla x, invoca a função “obtainCoordinates” com a opção para entrar na condição que permite, alterar a cor do quadrado para vermelho, registar num vetor o cubo selecionado, e invoca a função “saveCoordinates” que irá guardar num objeto as coordenadas do primeiro e segundo quadrado selecionado. Assim que as coordenadas do segundo quadrado sejam registadas, a função “lineMP” recebe esses pontos como argumento e trata de fazer o seu processamento o qual retorna um vetor de pontos, referentes aos pontos que formam a linha do ponto médio. De seguida a função “drawTilesLMP” e “drawLineMP” são invocadas, a primeira tem a responsabilidade de sobrepor aos cubos do tabuleiro os ladrilhos amarelos que são posicionados de acordo com as coordenadas da linha do ponto médio. Logo de seguida, a linha do ponto médio é desenhada utilizando as coordenadas do ponto inicial e final.

Caso a tecla pressionada seja a backspace, então será chamada a função “resetBoard” que irá tratar de remover todos as seleções, ladrilhos e linhas que tenham sido adicionados no tabuleiro. A função “cleanTilesLMP” trata de remover da cena todos os ladrilhos que tenham sido renderizados na cena, a função “cleanLineMP”, remove da cena a linha do ponto médio e por fim a função “resetCubeColor” restaura as cores iniciais de todos os cubos selecionados como ponto A e ponto B das linhas de ponto médio.

A função “lineMP” é a função responsável por processar a linha do ponto médio através das coordenadas do ponto A e ponto B que são recebidos como argumento. Esta começa por verificar as condições iniciais do algoritmo para determinar se a reta se encontra dentro da condição essencial, ou seja, se está no primeiro octante. Caso não esteja, aplica a transformação para a tornar numa reta do primeiro octante. Posteriormente se houve esta troca, é necessário inverter a transformação para a tornar no caso original. Após estes procedimentos a função retorna um vetor com as coordenadas que formam a linha do ponto médio.

Testes executados com o live Server do Visual Studio na porta 5500.

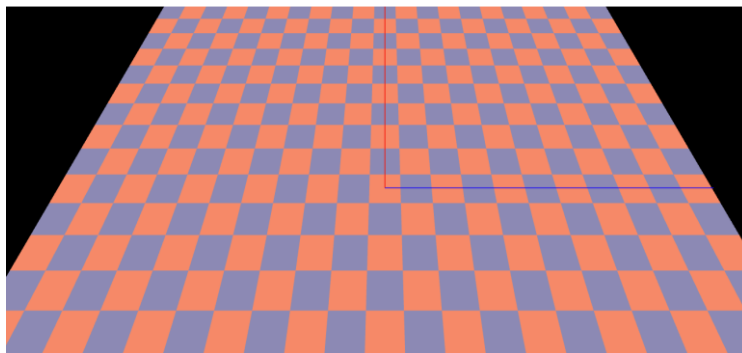


Figura 1 Posição inicial da aplicação com foco no ponto $(0,0)$

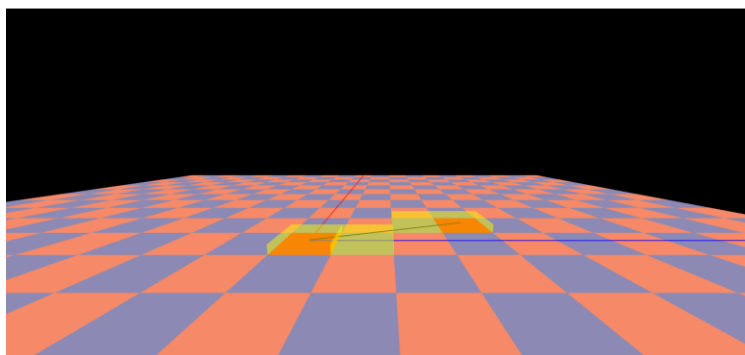


Figura 2 Selecção dos pontos $(0,0)$ e $(3,1)$, renderização dos tiles e da linha

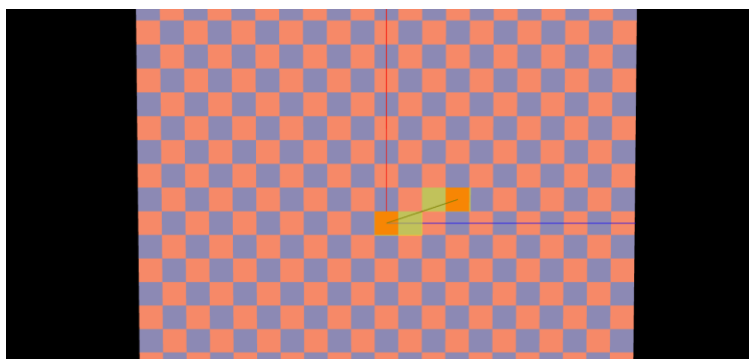


Figura 3 Vista do topo, liberdade do utilizador rodar o plano verticalmente

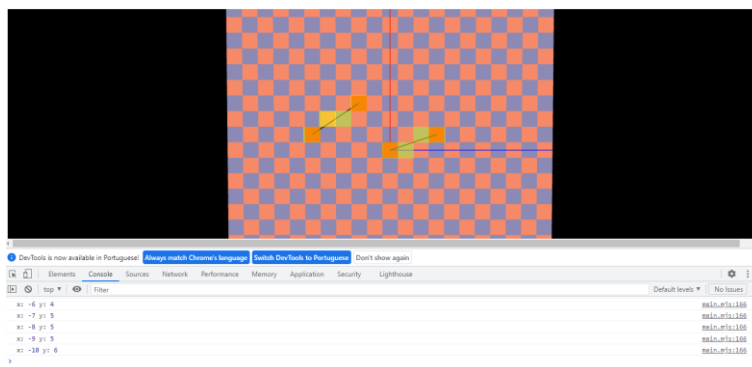


Figura 4 Apresentação das coordenadas na consola

Bibliografia:

Rasterização - <https://ler.letras.up.pt/uploads/ficheiros/artigo3521.pdf>

Documentação disponibilizada no âmbito da disciplina