Aluno: Renilton Ribeiro de Oliveira Matrícula: 11328395

Atividade 1.1

Introdução

Esse é um relatório sobre a primeira atividade de computação gráfica, implementação de um algoritmo para a rasterização de pontos e linhas, essa disciplina é ministrada pelo professor Christian Pagot, na UFPB, no período 2020.2. Para esse trabalho foi utilizada a linguagem de programação Javascript juntamente com o framework disponibilizado pelo professor que simula o acesso à memória de vídeo.

Bom, a rasterização nada mais é do que dado dois pontos extremos de um segmento de reta, determinar que pixels localizados entre eles devem ser plotados para compor o seguimento para ser desenhado na tela.

Estratégia

Optei pela a utilização do algoritmo de Bresenham, demorei alguns dias para conseguir utilizá-lo, primeiramente ele estava funcionado apenas para o primeiro octante, daí tive que entender como generalizá-lo para funcionar em todos os octantes, após fazer o de linhas funcionar para todos os octantes foi fácil o do triângulo já que só precisei passa os pontos finais e inicias e utilizar o algoritmo de linhas, já na parte da interpolação optei por modificar cada elemento do array individualmente, também tive que dedicar um tempo curto a conhecer algumas características da linguagem javascript, decidi por conveniência rodar todos os códigos na minha própria máquina, utilizando o visual studio code juntamente com a extensão line server para visualização em tempo real dos resultados de modificações no código.

Resultados

A funçãoMidPointLineAlgorithm recebe como parâmetros a posição dos dois pixels de coordenadas “x1y1 e x2y2”, e suas respectivas cores “color_0 e color_1”. Nessa função os valores de dX e dY são calculados, em seguida são realizadas diversas comparações e o resultado dependerá de acordo com os valores assumidos por dX e dY.

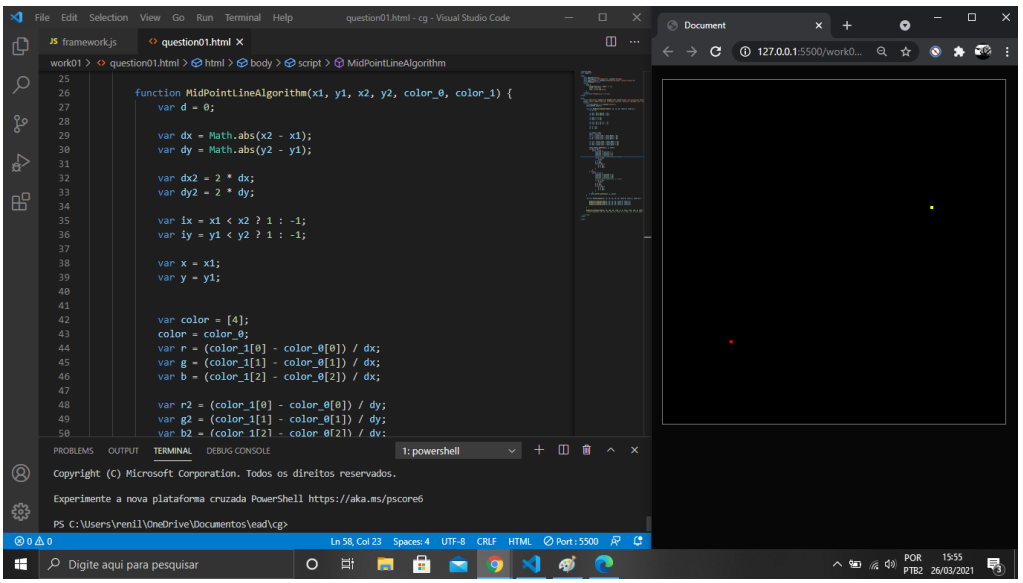

fig 1. Setando os primeiros pontos

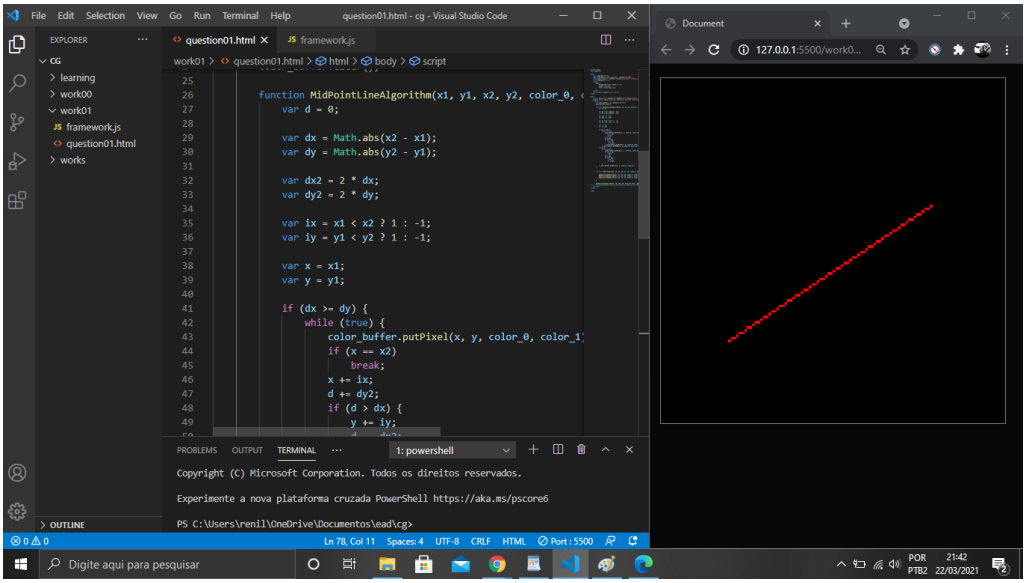

fig 2. Ligando os pontos por meio do algoritmo de rasterização

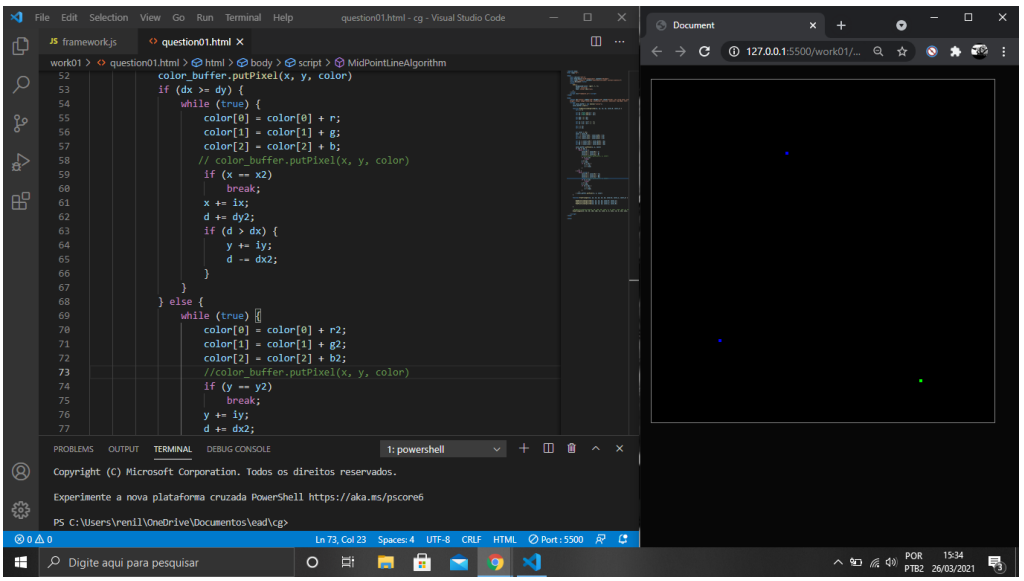

fig 3. sentando os 3 pontos do triângulo

Para rasterizar o triângulo dentro da função DrawTriangle chamo a função MidPointLineAlgorithm 3 vezes para desenhar suas linhas como visto na imagem.

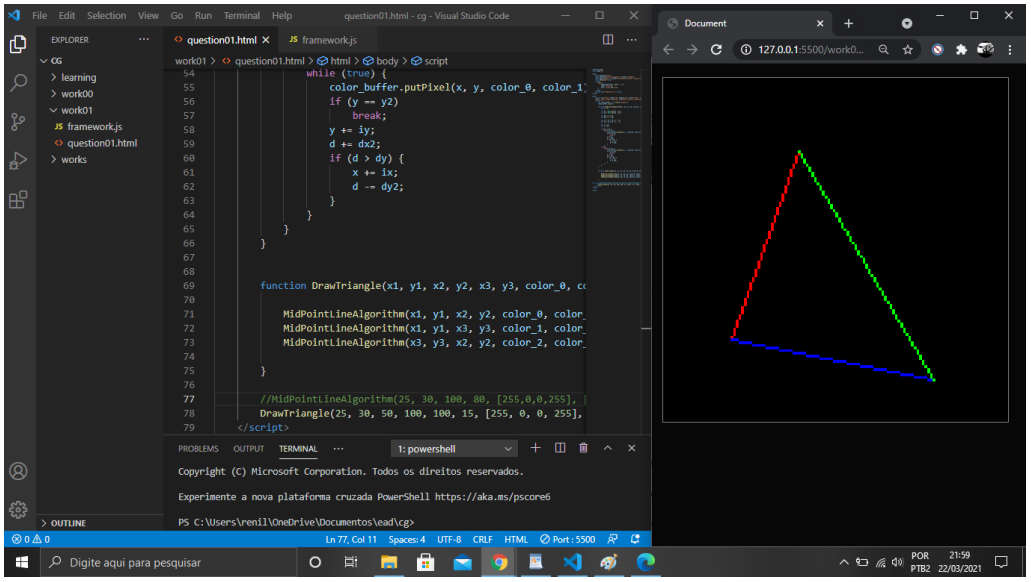

fig 4. Desenhando o triângulo por meio do algoritmo de rasterização

Interpolação de cores

O algoritmo nada mais faz do que interpolar a cor de um ponto final com a cor do ponto inicial fazendo um degradê suave entre eles. optei por pegar a diferença entre as duas cores e dividi pelo maior deslocamento da reta no eixo(dx ou dy) a depender do octante, isso em cada posição do array, por exemplo, `var r = (color_1[0] - color_0[0]) / dx;`, esse valor eu incremento na cor inicial em cada posição do array, por exemplo, `color[0] = color[0] + r;`

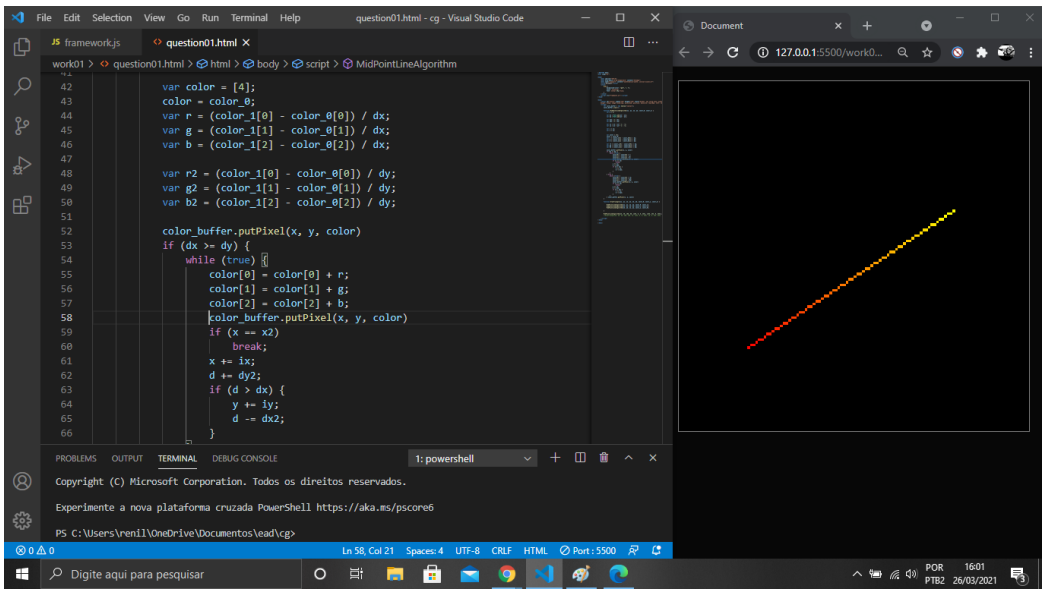

fig 5. Interpolando uma linha

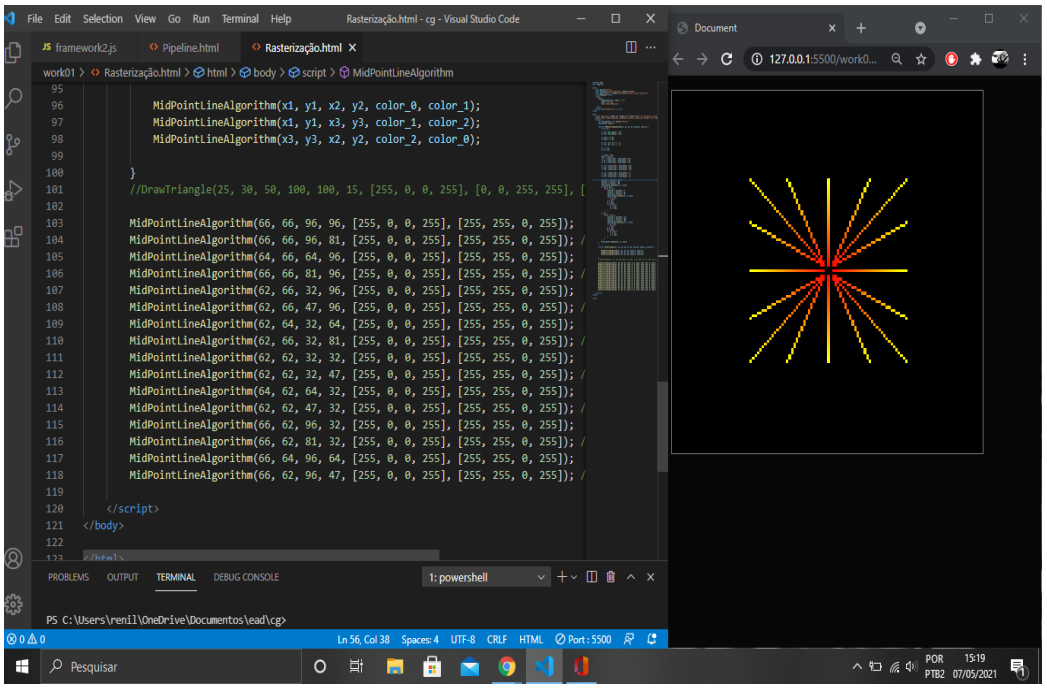

fig 6. Interpolando linhas em todos os octantes

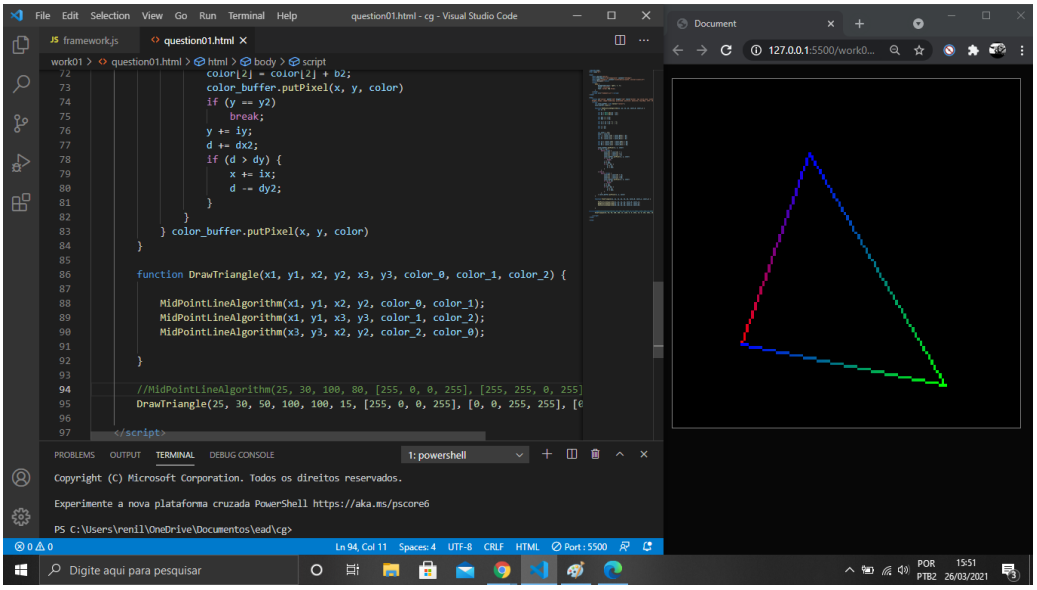

fig 7. interpolação de cores do triângulo

Dificuldade e Melhorias Possíveis

Tive dificuldade em entender o algoritmo de bresenham, como ele funciona em todos os octantes, além da parte de interpolação de cores, para entender que era apenas fazer a diferença das cores divido pelo dx ou dy a depender do octante que queremos fazer a interpolação.

O Trabalho poderia ser refeito em outra linguagem que o deixasse mais otimizado, poderia ter sido feito o desenvolvimento de funções que permitisse a alteração da inclinação das retas em tempo de execução, poderia ter sido feito preenchimento da figura (triângulo), melhor rasterização de cores de certa parte do triângulo, além disso, no meu código com certeza deve haver coisas que poderiam ter sido feitas de forma mais eficaz por falta de experiência de minha parte.

Referências

1. Notas de aula do Prof. Christian Azambuja Pagot .
2. The Bresenham Line-Drawing Algorithm, por Colin Flanagan. http://www.cs.helsinki.fi/group/goa/mallinnus/lines/bresenh.html
3. Bresenham's line algorithm https://en.wikipedia.org/wiki/Bresenham%27s_line_algorithm
4. http://matheuspraxedescg.blogspot.com/2016/08/trabalho-1-rasterizacao-de-ponto-e-linha.html?m=1
5. https://medium.com/@biancaamorimelo/rasteriza%C3%A7%C3%A3o-de-primitivas-em-opengl-8680a76fdda5
6. http://fleigfleig.blogspot.com/2016/08/interpolacao-de-cores-e-triangulos.html
7. http://thuliocg20151.blogspot.com/2015/05/interpolacao-de-cores.html

Link para o código

https://codepen.io/Reniwtz/pen/bGgGVGJ?editors=0010