

Link para o repositório do GitHub:

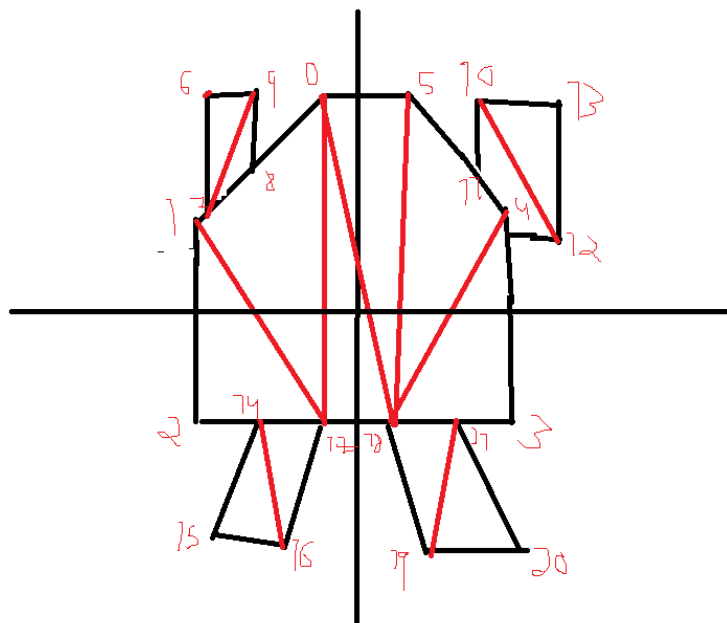
<https://github.com/VictorC98/abcg/tree/main/AtividadesPraticas/Atividade1>

Link para o Github Pages com o jogo:

<https://victorc98.github.io/abcg/Frogger/>

Técnicas utilizadas

O personagem jogável, o sapo, foi desenhado no jogo utilizando o identificador `GL_TRIANGLES` na função de renderização `glDrawArrays` para criar o conjunto de triângulos que compõem o personagem. Segue abaixo o esboço usado para planejar o posicionamento dos vértices e das arestas:



Durante todo o jogo, o sapo está fixado no centro de tela. Todos os outros objetos se movem conforme o jogador avança, assim dando a ilusão de que é o sapo que está se movendo. Os carros começam se movendo no eixo horizontal, em um sentido e uma velocidade aleatória. Quando um carro sai dos limites da tela, ele reaparece no lado oposto.

O fundo cinza e a lagoa também foram desenhados com `GL_TRIANGLES`. Os carros foram desenhados com o identificador `GL_TRIANGLE_FAN`, gerando os retângulos que representam os carros. Todos esses objetos do jogo armazenam o

arranjo de vértices na memória como um Vertex Buffer Object (VBO), que são atrelados a um Vertex Array Object (VAO).

A cor dos objetos é determinada pelas componentes vermelha, verde e azul (RGB) do vetor `m_color`, com cada componente tendo um valor entre 0 e 1. Por exemplo, o sapo tem a componente verde igual a 1, azul igual a 0.2 e vermelha igual a 0, deixando o sapo com uma cor verde clara.

O jogo utiliza alguns recursos, localizados no subdiretório “assets”. Os arquivos `objects.vert` e `objects.frag` contêm o código-fonte do vertex shader e fragment shader utilizados em todos os objetos. O vertex shader cuida do processamento dos vértices e o fragment shader processa cada fragmento individualmente após a rasterização. O arquivo `Freedom.ttf` é a fonte Freedom usada nas mensagens de vitória e derrota quando o jogo termina.

O controle do sapo é descrito pela estrutura `SDL_Event` da SDL, divididos em pressionamento de teclas (`SDL_KEYDOWN`) e liberação de teclas (`SDL_KEYUP`). As animações do sapo e dos carros utilizam a função `getDeltaTime` para que elas sejam sincronizadas com o tempo. As colisões entre os objetos são detectadas usando a distância euclidiana entre as coordenadas de translação dos objetos.