

# Relatório do Projeto: Algoritmo de Preenchimento de Polígonos (ET/AET)

Lucas Greff Meneses - 13671615  
Vitor da Silveira Paula - 10689651

29 de setembro de 2025

## Participação dos Integrantes

- **Lucas:** Implementação da interface gráfica em PyQt6 e integração com OpenGL.
- **Vitor:** Implementação do algoritmo de preenchimento utilizando as tabelas ET e AET, testes com polígonos simples e complexos e elaboração do relatório.

## 1 Introdução

O objetivo deste projeto foi implementar o algoritmo de preenchimento de polígonos baseado na coerência de arestas, utilizando as estruturas **ET** (Tabela de Lados) e **AET** (Tabela de Lados Ativos).

A implementação foi realizada em **Python**, utilizando as bibliotecas *PyQt6* para a interface gráfica e *PyOpenGL* para a renderização. O sistema desenvolvido permitirá, em trabalhos futuros, a integração de técnicas mais avançadas de rendering em OpenGL.

## 2 Metodologia

O sistema foi estruturado em duas partes principais:

### 2.1 Interface Gráfica

A interface gráfica foi desenvolvida com *PyQt6*, permitindo ao usuário:

- Inserir pontos do polígono através de cliques do mouse.
- Fechar o polígono e visualizar sua borda.
- Alterar a espessura do traçado e a cor de preenchimento.
- Executar o algoritmo de preenchimento.
- Limpar a tela para criar novos polígonos.

## 2.2 Algoritmo de Preenchimento (ET/AET)

O algoritmo de varredura por linhas (*scanline fill*) foi implementado seguindo os seguintes passos:

1. Construção da **Tabela de Lados** (ET), que organiza as arestas por linha de início.
2. Atualização da **Tabela de Lados Ativos** (AET) a cada linha de varredura.
3. Cálculo das interseções da scanline com as arestas ativas.
4. Ordenação das interseções e preenchimento dos segmentos entre pares de pontos.

## 3 Resultados

O sistema foi testado em diferentes cenários:

- **Polígonos convexos:** preenchimento correto em todas as linhas de varredura.
- **Polígonos côncavos:** o algoritmo conseguiu identificar e preencher corretamente as regiões internas.
- **Polígonos complexos:** para polígonos não auto-intersectantes, o preenchimento também foi válido.

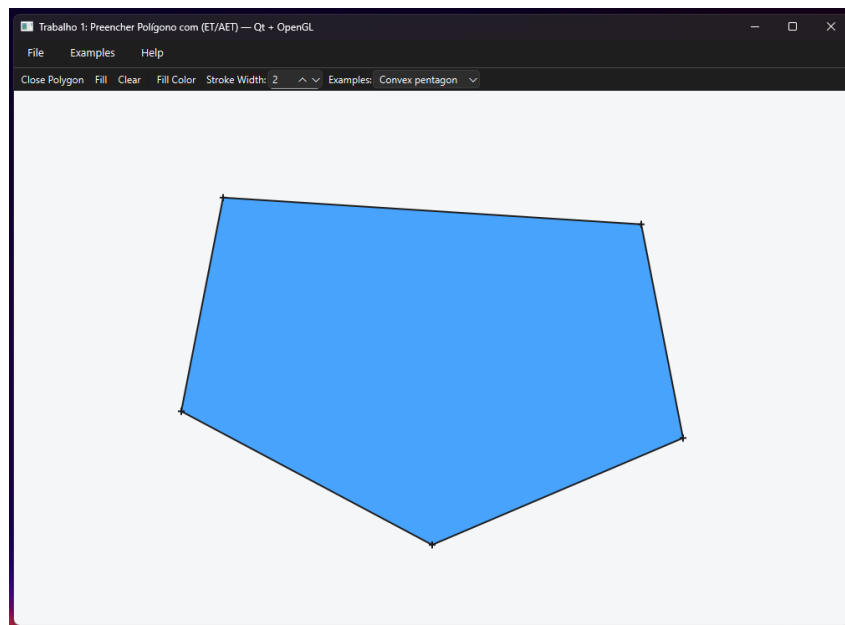


Figura 1: Exemplo de polígono preenchido pelo sistema (teste com polígono côncavo).

## 4 Conclusão

O projeto permitiu aplicar conceitos fundamentais de computação gráfica, unindo a implementação do algoritmo de preenchimento (ET/AET) com a construção de uma interface interativa em Qt e OpenGL.

Os resultados confirmaram a validade do algoritmo para polígonos simples e complexos, atendendo aos requisitos propostos. O sistema está pronto para servir como base na implementação de técnicas mais avançadas de *rendering* em OpenGL.