

Computação Gráfica

Trabalho 2

Aluno: João Vitor Branquinho Ribeiro

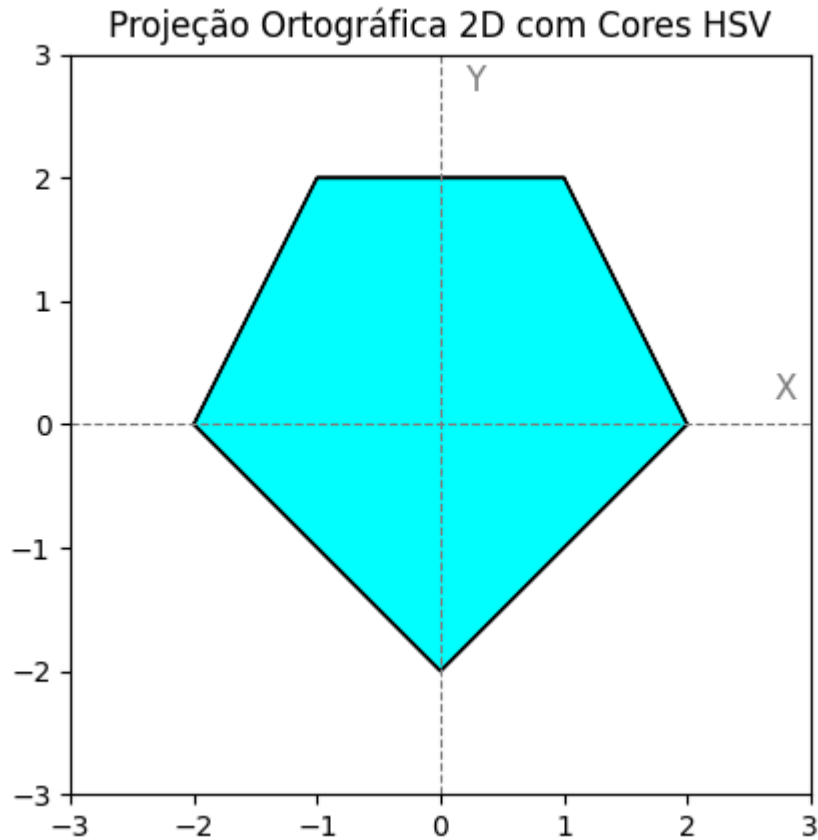
Matrícula: 216.031.144

1- Para cada face 'i' do objeto, foi atribuída uma cor no espaço HSV com **saturação (S)** e **valor (V)** máximos, ambos iguais a 1. A **matiz (hue)** foi determinada pela fórmula $h=360 \ i/(n+1)$.

Aqui está como foi feito:

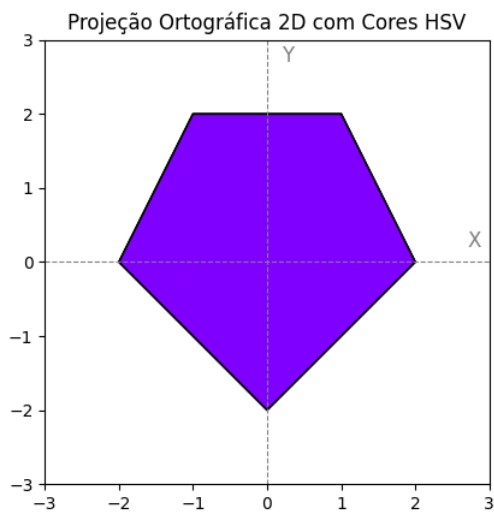
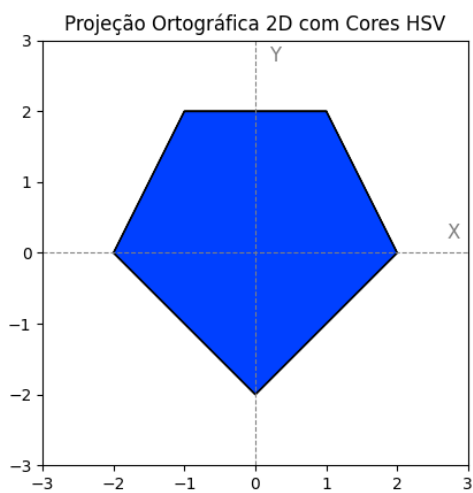
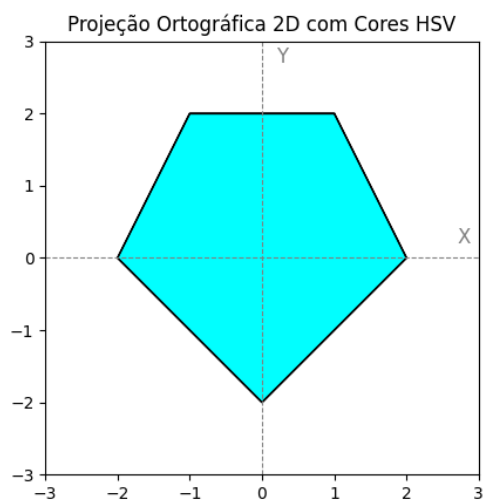
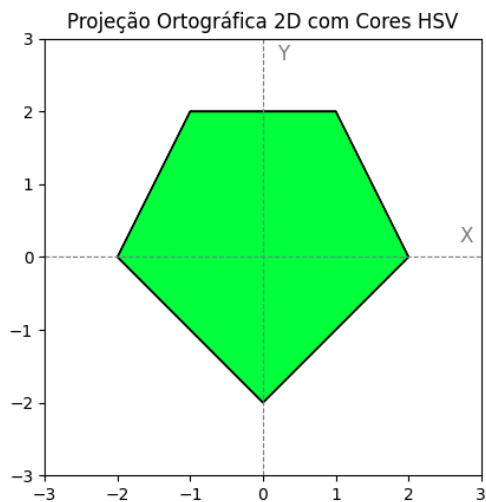
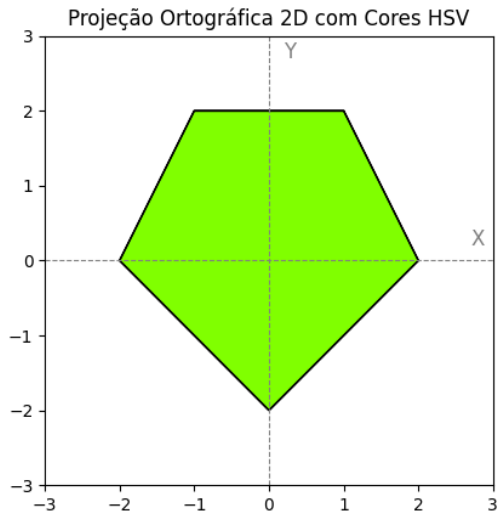
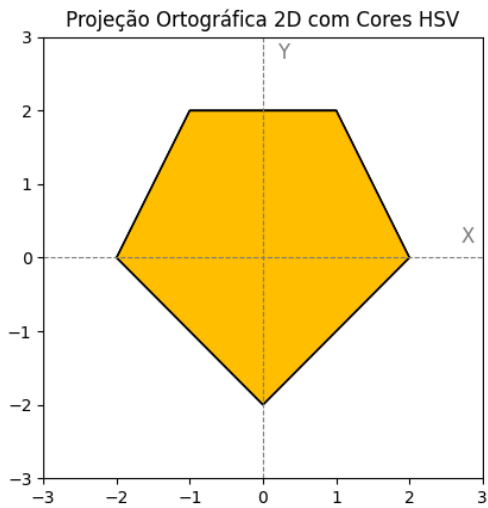
```
21 faces = [  
22     [0, 1, 2, 3, 4],           # base  
23     [5, 6, 7, 8, 9],         # topo  
24     [0, 1, 6, 5],            # lateral 1  
25     [1, 2, 7, 6],            # lateral 2  
26     [2, 3, 8, 7],            # lateral 3  
27     [3, 4, 9, 8],            # lateral 4  
28     [4, 0, 5, 9],            # lateral 5  
29 ]  
30  
31 # ===== GERAR CORES HSV E CONVERTER PARA RGB =====  
32 qtd_faces = len(faces)  
33 hsv_colors = []  
34 for i in range(qtd_faces):  
35     h = (360 * i) / (qtd_faces + 1)  
36     h_norm = h / 360.0 # normalizar para o intervalo [0, 1]  
37     rgb = colorsys.hsv_to_rgb(h_norm, 1.0, 1.0) # S = 1, V = 1  
38     hsv_colors.append(rgb) # convertemos para rgb  
39 print(hsv_colors)
```

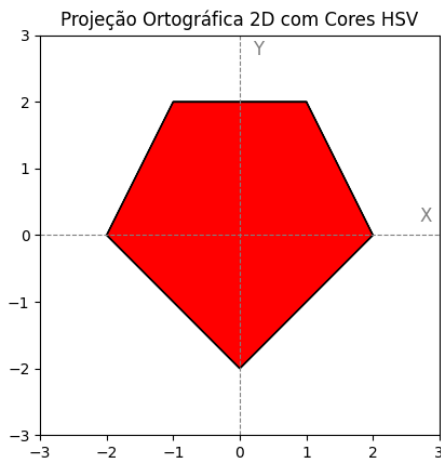
2-Para a projeção, decidi usar a **projeção paralela ortográfica**, em que pego meu polígono e “descarto” um eixo (no meu caso, o eixo Z).



Vale observar que estão sendo desenhadas 2 faces! Porém, por elas estarem uma sobre a outra, apenas a desenhada por última aparece. Esse era o efeito esperado, pois essas faces possuem as mesmas coordenadas (x,y), apenas se diferenciando na coordenada Z (que estamos descartando).

3-Para o enunciado 3, irei mostrar as cores que estão sendo pintadas nas faces.





4- Para conseguir implementar o enunciado 2 na prática, foram utilizados muitos pontos do capítulo 1 e 2 do livro.

Com os assuntos do capítulo 1, consegui chegar a uma definição do sólido que eu queria criar utilizando de estruturas de dados de uma linguagem de programação (python) para representar as estruturas de dados de um polígono, como vértices, arestas e faces. Além disso, verifico se meu objeto é realizável por meio da fórmula de Euler.

Com os assuntos do capítulo 2, consigo ter um sistema de coordenadas para definir meu polígono dentro de um programa. Além disso, aplico determinadas transformações geométricas por meio de matrizes de transformação, a fim de projetar meu polígono 3d em um plano 2d, para atingir a projeção desejada.

Nesse caso, a projeção final é uma projeção ortográfica, podendo ser simplificada em apenas “descartar o eixo Z”.