

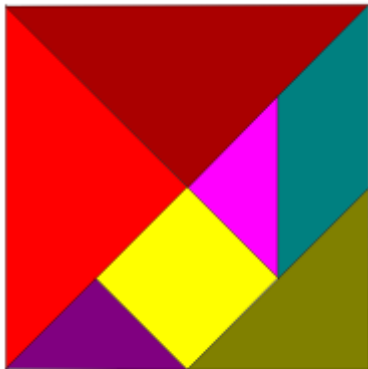
## CCI-36 / CMC-30 - ano de 2022

**Prof. Carlos Henrique Forster**

### Projeto 1 - Tangram

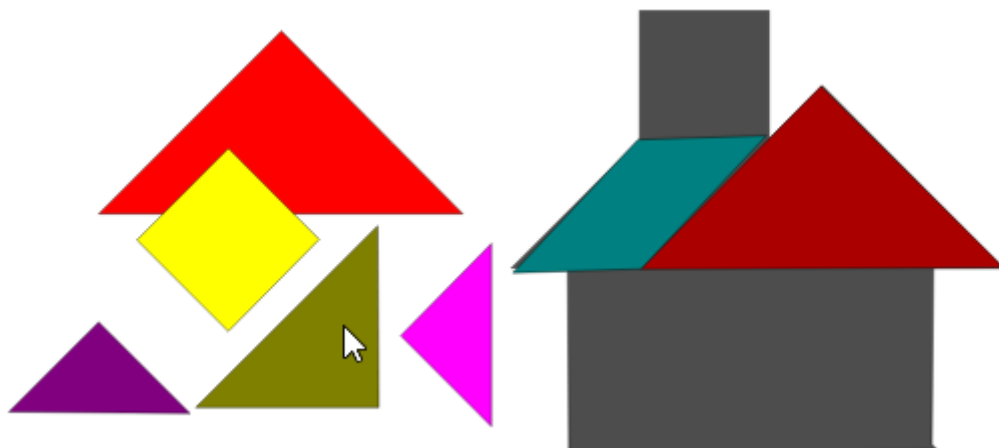
Foi dada de exemplo uma página web com um SVG interativo.

Nele pode-se arrastar as peças do tangram e colocar sobre o template.



Quando pressionado sobre uma das peças, o movimento do mouse arrasta a peça consigo, até que o botão seja solto.

Colocando o mouse sobre uma peça, a roda do mouse permite a rotação da peça. Assim, é possível ir movendo as peças até preencher o template.



O jogo (ou fase) é completado quando toda a região cinza for coberta (ou aproximadamente para não abusar muito do usuário).

No exemplo, pode-se observar a lógica do tratamento dos eventos do mouse: "mousedown", "mouseenter", "mouseleave", "mousemove", "mouseup" e "mousewheel" e das variáveis de estado: posições e rotações das peças, qual peça está sob o cursor do mouse, qual peça está sendo arrastada, qual foi a posição em que a peça foi "agarrada".

A tarefa consiste em refazer esse jogo (não precisa ser idêntico) utilizando ou HTML5/CANVAS ou THREE.JS

Pede-se para implementar e explicar em relatório os seguintes pontos:

- função **point-in-polygon** que retorna se um ponto está dentro de um polígono. Isto vai permitir saber se o cursor do mouse está sobre algum polígono.
- função **polygon-intersection-area** que retorna a área de intersecção de dois polígonos (ou uniões de polígonos). Isto vai permitir sabermos o quanto falta para cobrir a área cinza e também se duas peças tenham uma intersecção significativa.
- tratamento de eventos semelhante ao do exemplo, mas avisar o usuário que o jogo ou fase terminou, entre outras melhorias possíveis.
- resolver o problema da rotação, criar uma forma mais conveniente para o usuário fazer a rotação. Por exemplo, será que apenas com o arrastar do mouse já não daria para definir a rotação?
- explicar os cuidados necessários com as transformações geométricas 2D em uso.