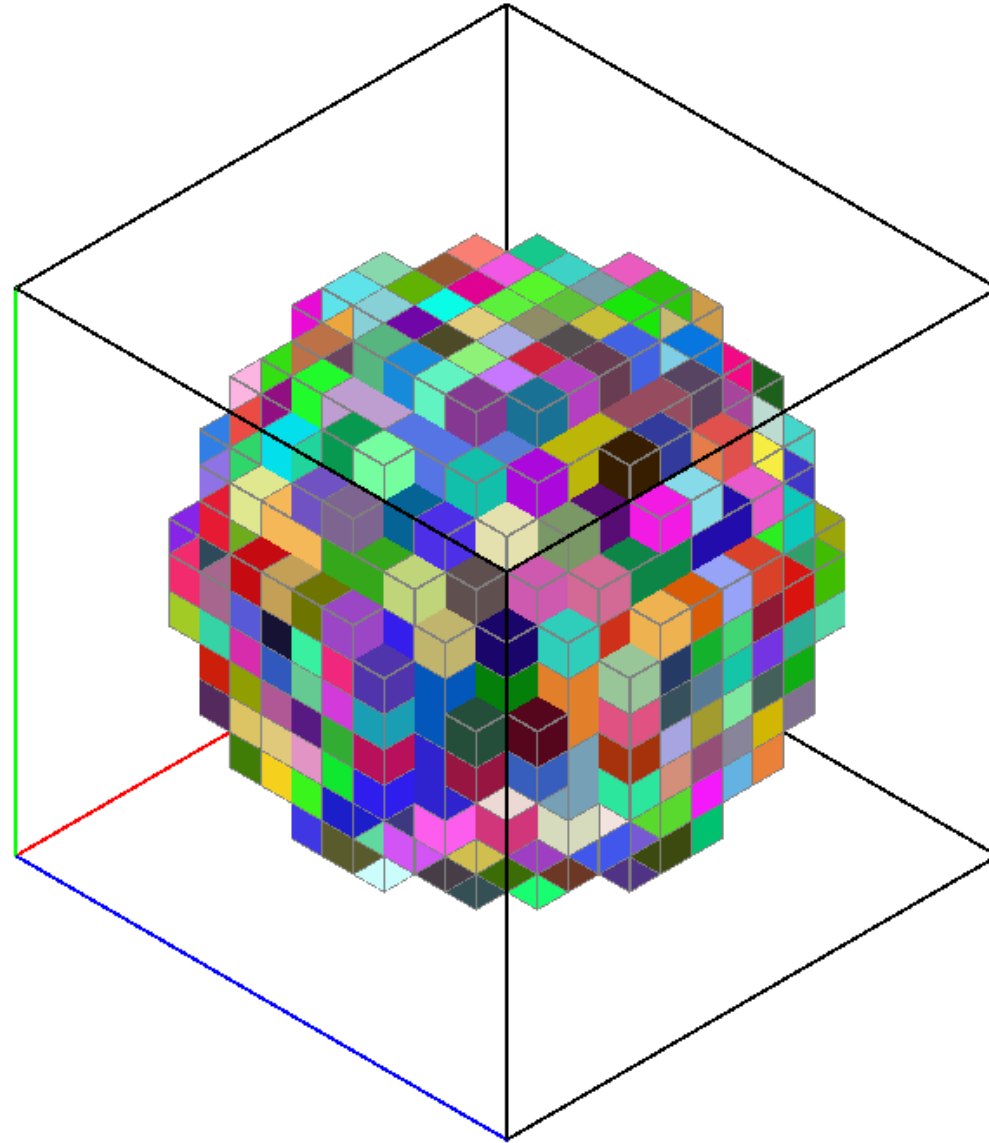


# Sickle



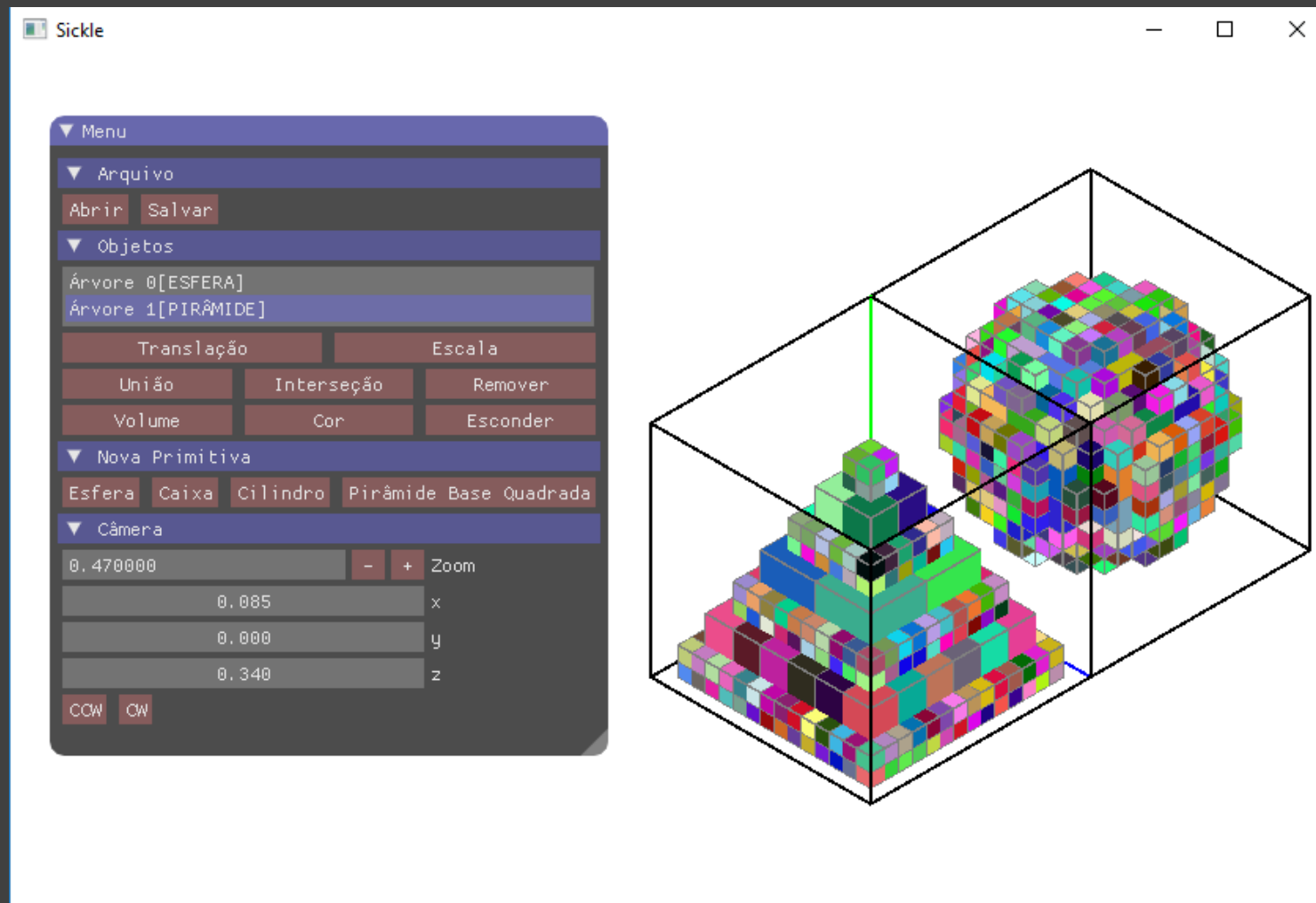
Carol Herbst – 354044

354047 – Daniel Augusto

# ▼ Introdução

Sickle é o modelador feito para o primeiro trabalho de modelagem.

Todos os modelos são representados como Octrees num espaço global.



# ▼ Ações Implementadas

Com cada modelo, podemos aplicar as seguintes funções:

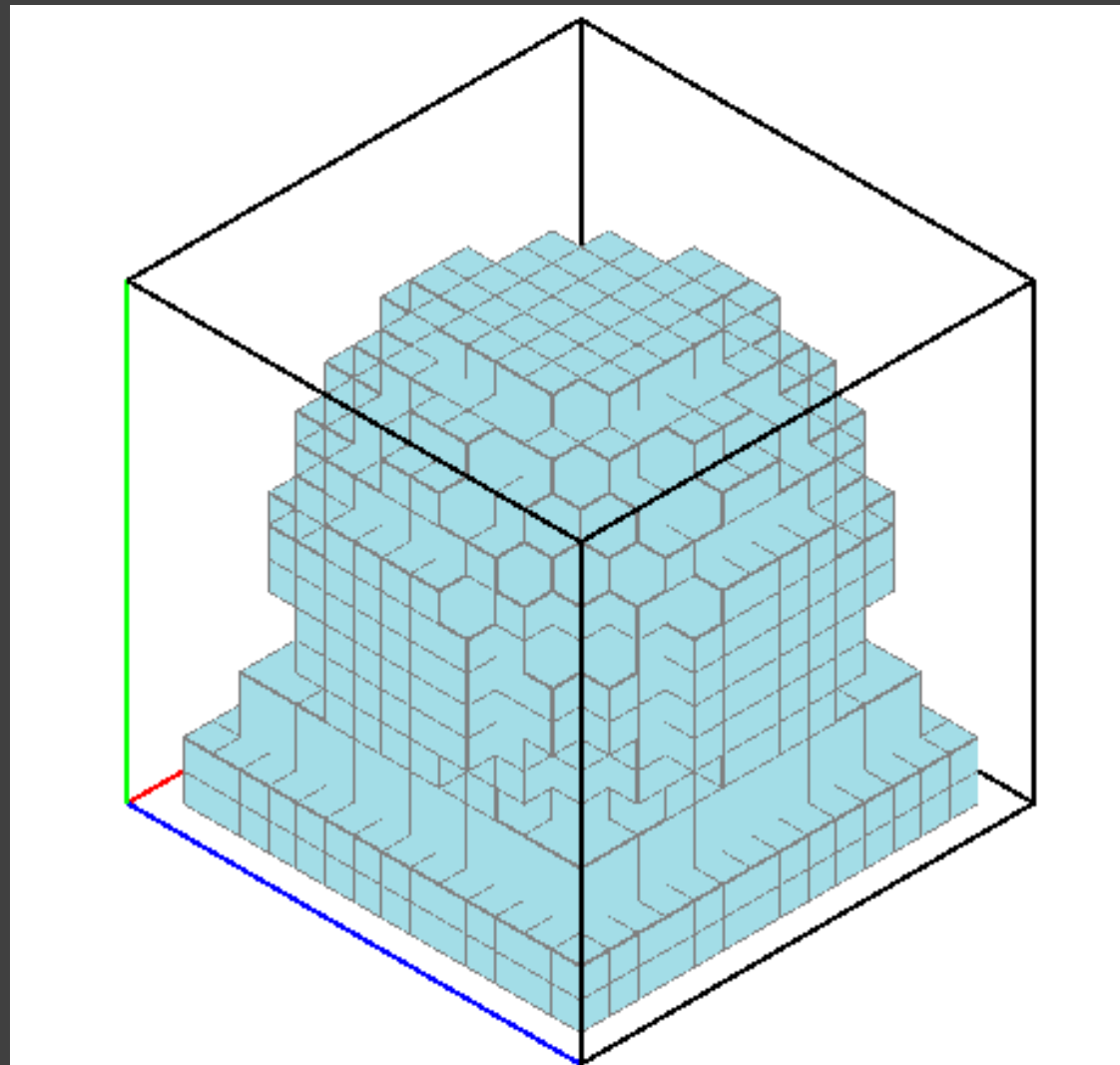
1. Translação e escala no espaço global.
2. União e intersecção de octrees de mesmo espaço local.
3. Mudanças de visibilidade e cor.
4. Cálculo de volume.



# ▼ União

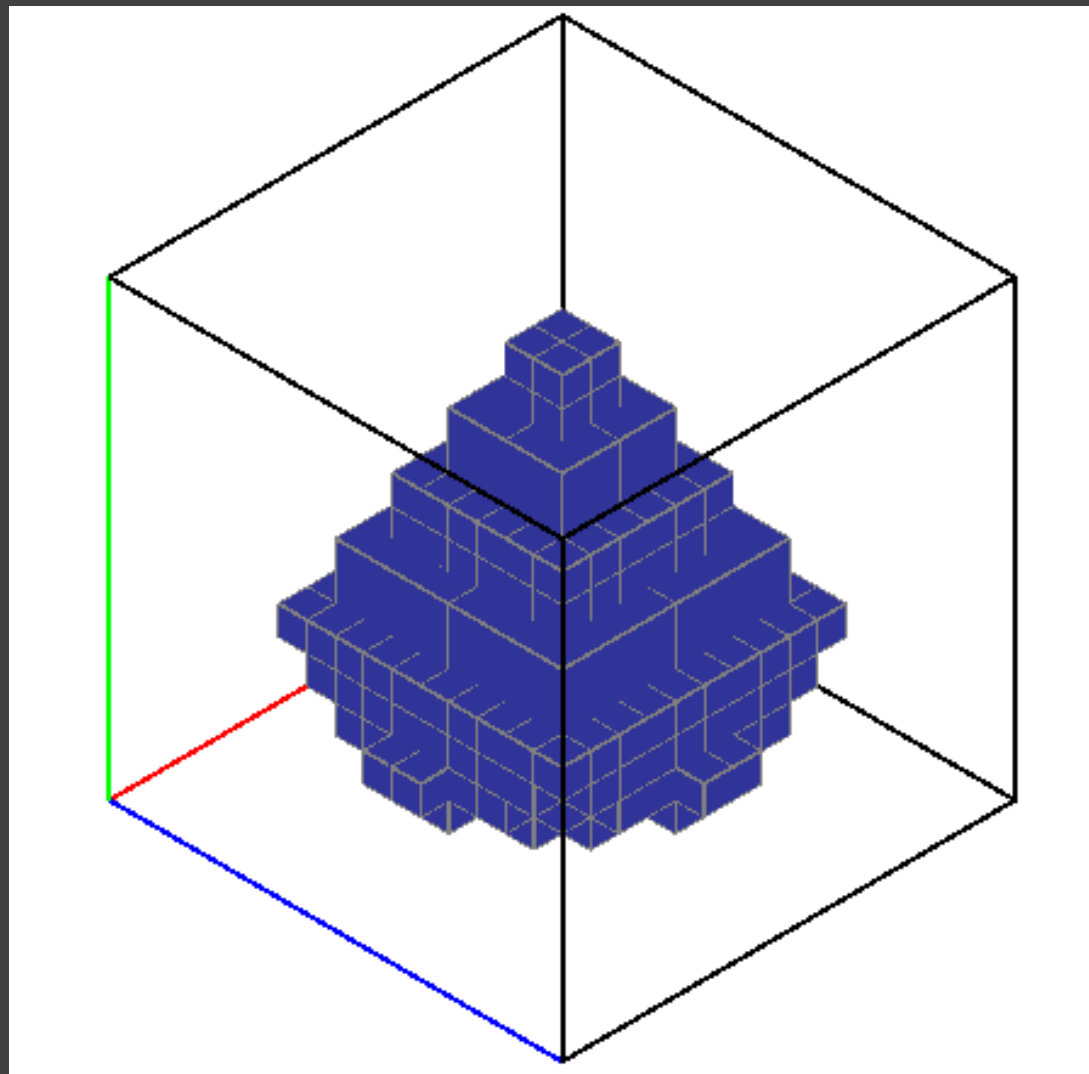
Aqui podemos ver a união da esfera e da pirâmide mostradas nos slides anteriores.

Usando a função de volume podemos ver que nosso sólido tem  $0.501953u^3$



# ▼ Intersecção

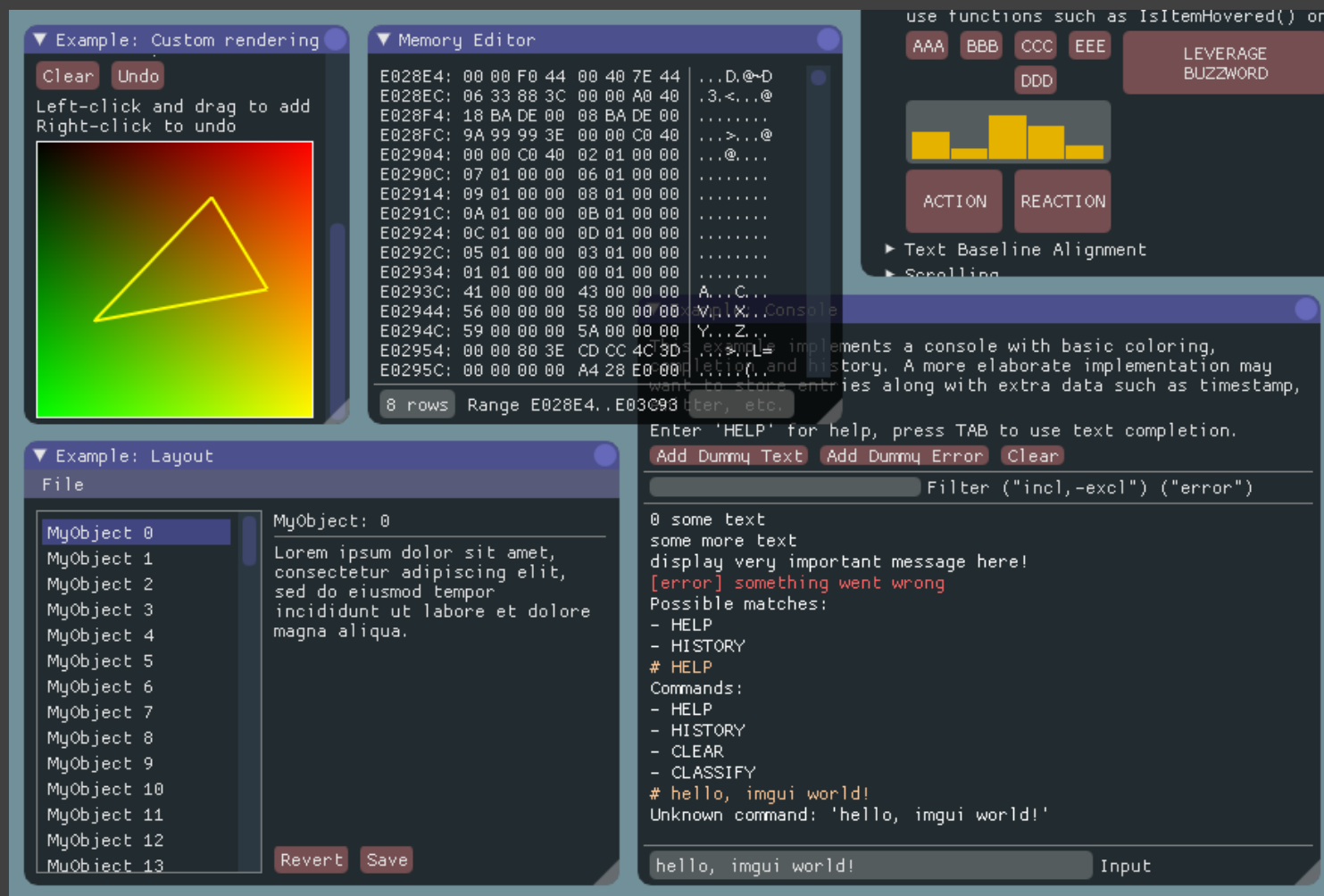
Aqui podemos ver a intersecção da esfera e da pirâmide mostradas nos slides anteriores.



# ▼ Bibliotecas Usadas

Na implementação do nosso modelador usamos as bibliotecas:

- OpenGL
- GLFW
- GLM
- Dear ImGui



# ▼ Immediate Mode Graphical User Interface

## C++ code

```
ImGui::Text("Hello, world %d", 123);

if (ImGui::Button("OK"))
{
    // do stuff
}

ImGui::InputText("string", buf, 256);

ImGui::SliderFloat("float", &f, 0.0f, 1.0f);
```

## Output Interface

### ▼ Debug

Hello, world 123

OK

Quick brown fox

string

0.600

float

## ▼ Classes usadas

- Model
  - Interface que representa um modelo qualquer.
- Octree
  - Implementa a interface Model, guarda uma árvore e uma bounding box.
- No futuro, mais tipos de modelos (Árvore CSG, por exemplo) irão implementar a mesma interface e poderão interagir entre si.



## ▼ Classes usadas

- Tree
  - Uma estrutura que representa o nó de uma árvore.
  - Se a árvore é branca, então usaremos um ponteiro nulo para representa-la.
  - Implementa todos os algoritmos recursivos.
    - Idealmente, eles seriam implementados de forma iterativa usando uma pilha em Heap para evitar stack overflow.

## ▼ Classes usadas

- Bounding Box
  - Sua função de desenho é usada no desenho da Octree.
  - É parâmetro de muitas funções de Tree.
  - É usado na classificação de objetos.

## ▼ Classes usadas

- Classifier
  - Define uma interface para funções, lambdas e classes que desejam representar uma função de classificação.
  - Para uma dada bounding box, retorna preto, cinza ou branco para representar cheio, parcial e vazio, respectivamente.

# ▼ Exemplo de classificador

```
Color Sphere::operator()(BoundingBox bb){
    if (sphere_box_intersection(center, radius, bb.getCenter(), bb.depth, bb.depth, bb.depth)) {
        size_t count = 0;
        for (size_t i = 0; i < 8; ++i)
            if (distance(center, bb.getVertice(i)) < radius)
                count++;
        if (count >= 8)
            return Color::black;
        else
            return Color::gray;
    } else
        return Color::white;
}
```

## ▼ Classes Usadas

- Sphere, Box, Cilinder e Square Pyramid
  - Classes que representam as primitivas do modelador. Recebem os parâmetros relevantes e implementam a interface Classifier
- Helper
  - Não é uma classe, e sim um arquivo de definições (.h) com algumas funções úteis para o sistema, como testes de interseções e criação de matriz de projeção

## ▼ Classes Usadas

- MainMenu
  - Implementa toda a interface utilizando o paradigma Immediate Mode GUI.
  - Possui a lista de modelos e a câmera.
  - Responsável por criar, deletar e invocar operações nos modelos e alterar os parâmetros da câmera

## ▼ Classes usadas

- Main
  - Inicializa as bibliotecas
  - Cria e carrega a projeção no OpenGL
  - Chama a função de desenho em cada modelo da lista
  - Implementa os callbacks de teclado e redimensionamento de janela

## ▼ Classes Usadas

- Sphere, Box, Cilinder e Square Pyramid
  - Classes que representam as primitivas do modelador. Recebem os parâmetros relevantes e implementam a interface Classifier
- Helper
  - Não é uma classe, e sim um arquivo de definições (.h) com algumas funções úteis para o sistema, como testes de interseções, e criação de matriz de projeção



## ▼ Classes usadas

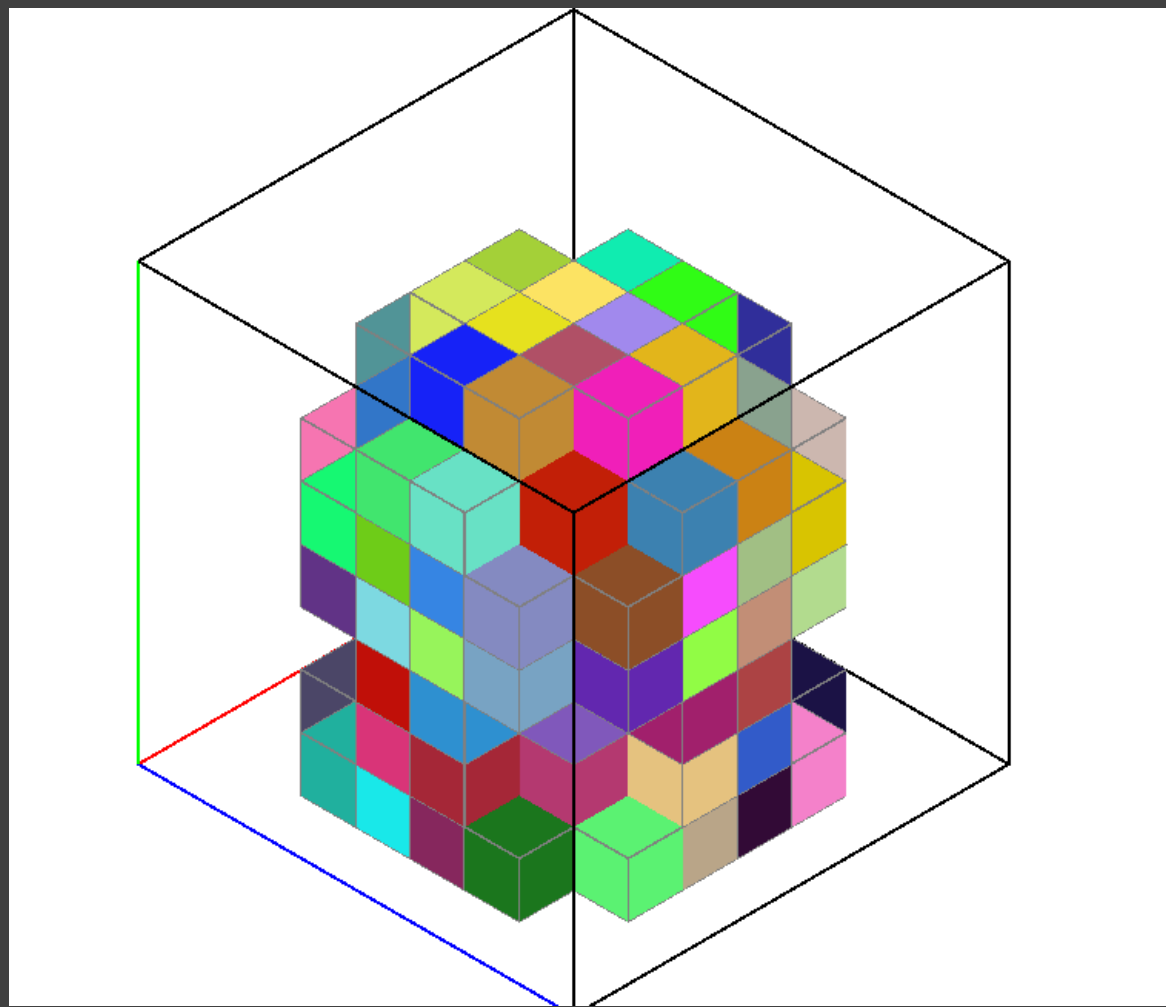
- MainMenu
  - Implementa toda a interface utilizando o paradigma Immediate Mode GUI.
  - Possui a lista de modelos e a câmera.
  - Responsável por criar, deletar e invocar operações nos modelos e alterar os parâmetros da câmera

## ▼ Classes usadas

- Main
  - Inicializa as bibliotecas
  - Cria a matriz de projeção utilizando as informações da câmera
  - Chama a função de desenho de cada modelo da lista
  - Implementa os callbacks de teclado e de redimensionamento de janela

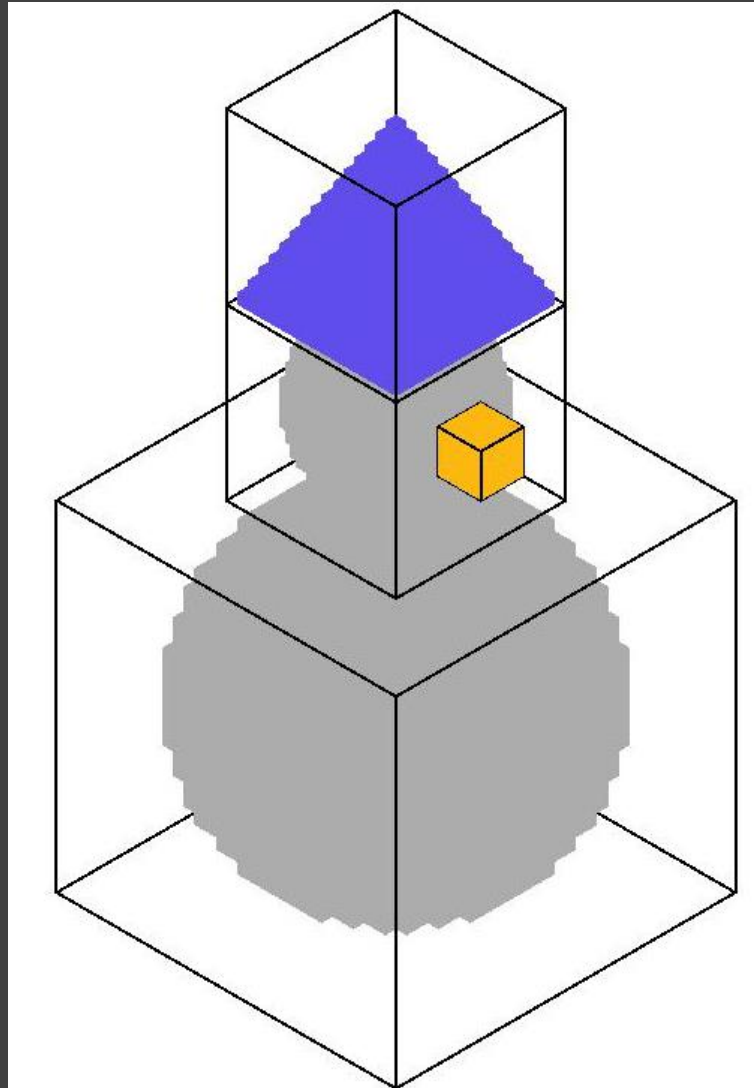
# ▼ Exemplos

União esfera e  
meia esfera



# ▼ Exemplos

Boneco de neve



# ▼ Exemplos

Nosso Tema:  
Catedral  
Metropolitana de  
Fortaleza



# ▼ Exemplos

Nosso Tema:  
Catedral  
Metropolitana de  
Fortaleza

