

# Manipuladores de transformações

por Julio Kenji Ueda 9298281

Para instalar todas das dependências, execute:

```
$ pip3 install -r requirements.txt
```

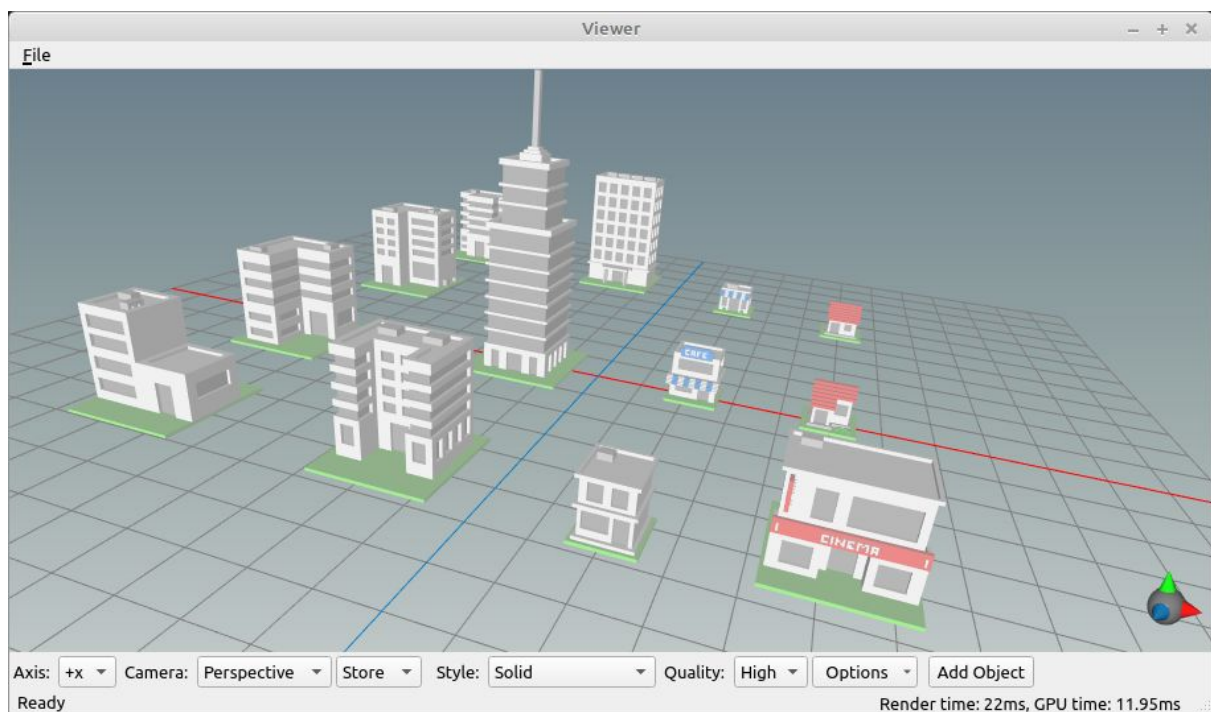
Para iniciar o programa, execute:

```
$ python3 main.py
```

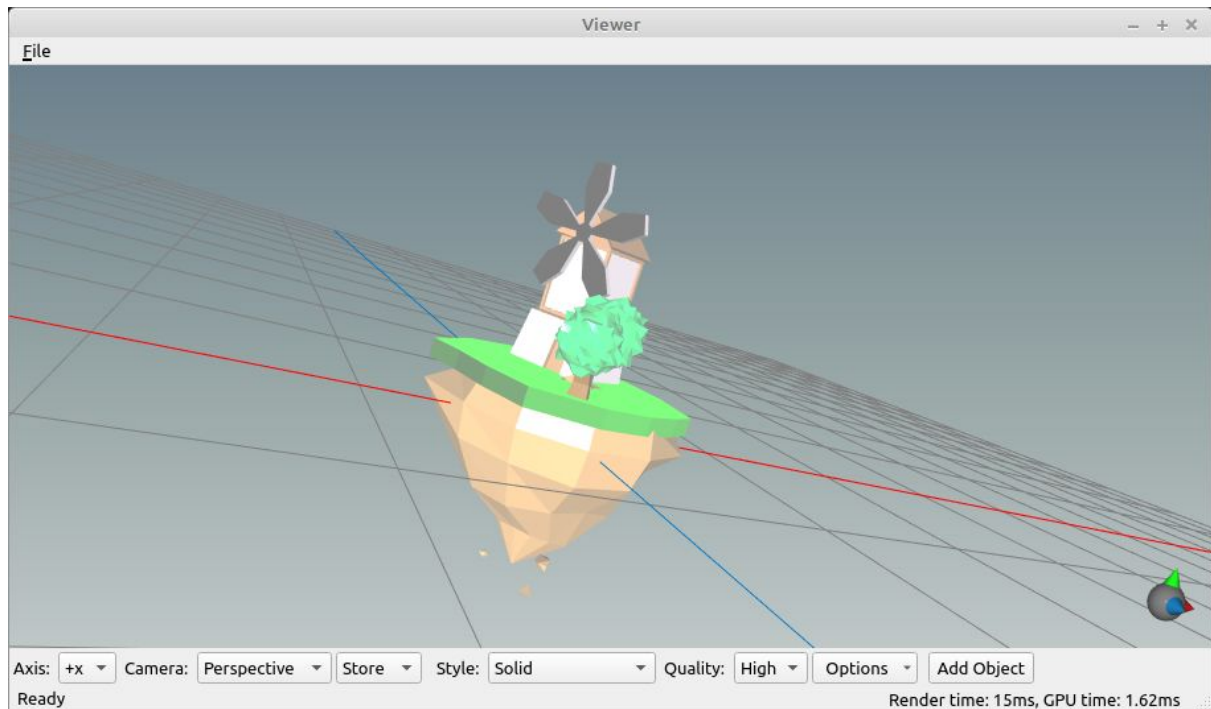
## Soluções

### Carregamento de arquivos OBJ

Para a leitura dos arquivos foi utilizada a biblioteca [PyWavefront](#), que importa os vértices de coordenadas, normais e de textura de cada material do objeto. Como cada Ator (classe **Actor**) possui um tipo de material associado, foi criada a classe **WFOParts** (subclasse de **Actor**) que representa a geometria de um material do objeto importado. O conjunto de todos os materiais do objeto é representado pela classe **WFObj** (subclasse de **Group**). As duas classes encontram-se na subpasta **/Source/Graphics**.



A biblioteca [PyWavefront](#) não permite o carregamento dos vértices com formatos diferentes em um único arquivo (isto é, conjunto misto de vértices com e sem normais, com e sem texturas), portanto não é possível carregar o arquivo **low-poly-mill.obj** nativamente. Para contornar este problema, a classe **Parser** foi criada para dividir o arquivo original em sub-arquivos de formatos iguais e interpretáveis pela biblioteca. A classe **Parser** encontra-se na pasta raiz do projeto.



O objeto low-poly-mill é composto por diversos atores de vários sub-arquivos

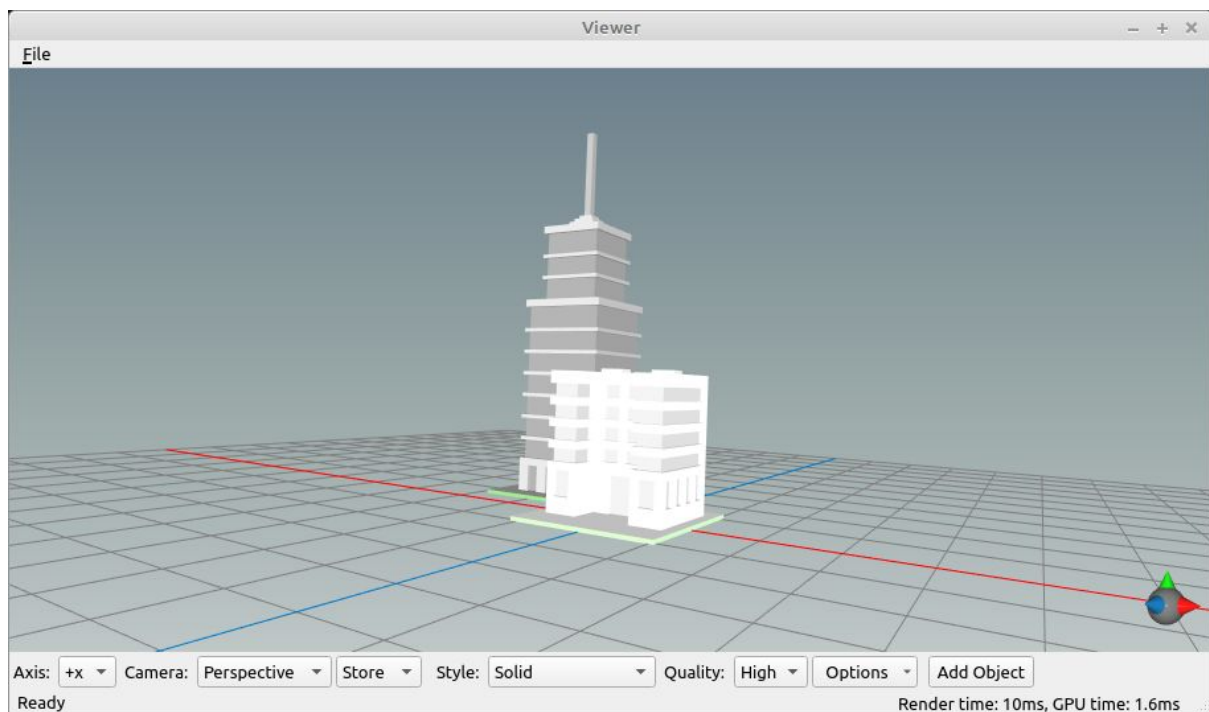
Para adicionar um objeto à cena, basta clicar no botão **Add Object** no lado direito do painel, que faz a chamada do método implementado **add\_object** da classe **Renderer**.

A biblioteca [PyWavefront](#) transforma as faces quadriláteras em faces triangulares.

## Seleção

A seleção dos objetos é obtida através do método implementado **\_get\_near\_obj** da classe **Renderer**, que emite um raio virtual que atravessa a cena a partir da câmera e retorna o objeto intersectado mais próximo. Se pressionado a tecla **<Shift>**, o método implementado **\_select\_object** seleciona este objeto (classe **WFObj**, subclasse de **Group**) que realiza a chamada do método **setHighlighted(True)** para cada parte (classe **WFOParts**, subclasse de **Actor**) e então este objeto aparece ressaltado. O programa então entra em modo de edição.

Ao clicar em um ponto da cena onde não existem objetos (quando **\_get\_near\_obj** não retorna nenhum objeto) ou quando o usuário pressiona a tecla **<ESC>**, o programa sai do modo de edição.



Somente o objeto mais próximo é selecionado

## Operações e transformações

As operações de translação e escala são realizadas com a ajuda do método implementado **slider** da classe **Renderer**, que rastreia o movimento do *mouse* e traduz em transformações incrementais (deslizamento para direita) ou decrementais (deslizamento para esquerda) de acordo com a operação e eixo escolhido.

A operação de rotação faz uso do método implementado **obj\_trackball** da classe **Renderer**, que mapeia os movimentos do *mouse* em rotações ao redor do centro do objeto. Movimentos verticais realizam a rotação em torno do eixo **X** do objeto e movimentos horizontais realizam a rotação em torno do eixo **Y** do objeto.

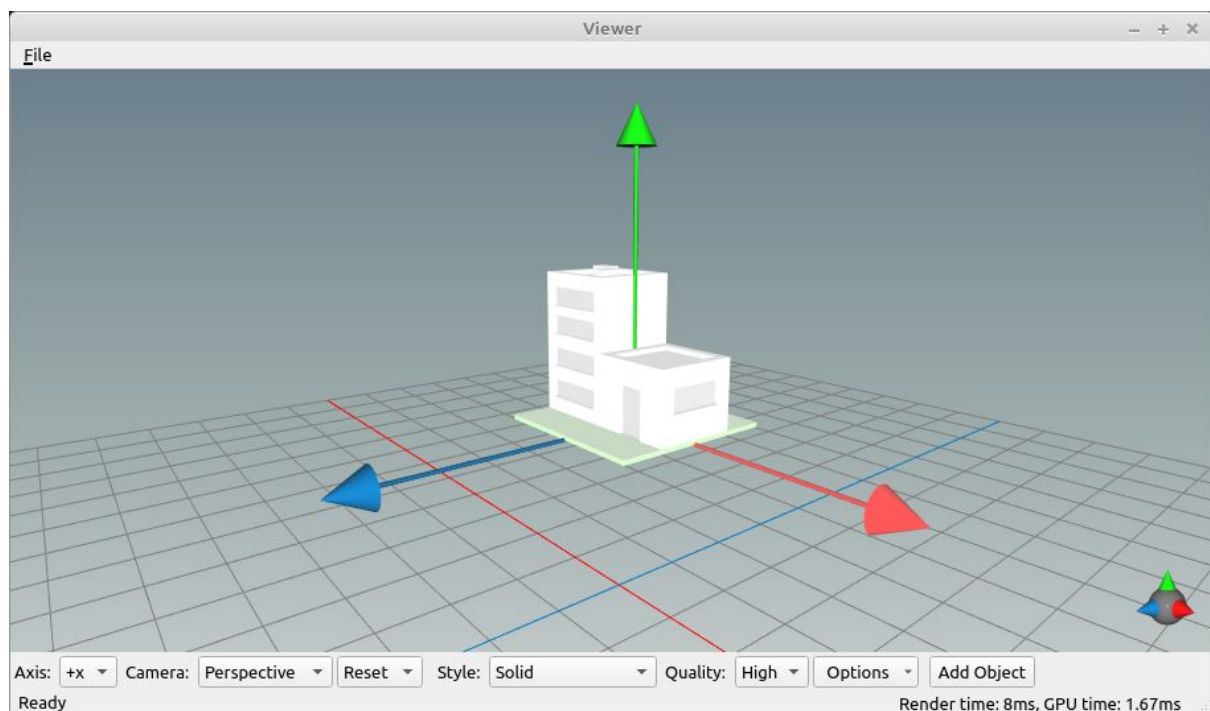
Ao clicar com o *mouse* sobre o objeto no início de qualquer operação, é guardada uma cópia do estado da matriz de transformação. Se no meio da operação o usuário pressionar **<ESC>**, a cópia da matriz é restaurada e a transformação é desfeita.

Para remover o objeto selecionado da cena basta pressionar a tecla **<Delete>** em qualquer situação. Para remover o objeto com a tecla **X** é preciso que o objeto apenas esteja realçado (sem o manipulador de transformação visível), caso contrário a tecla apenas irá realçar o eixo **X** do manipulador.

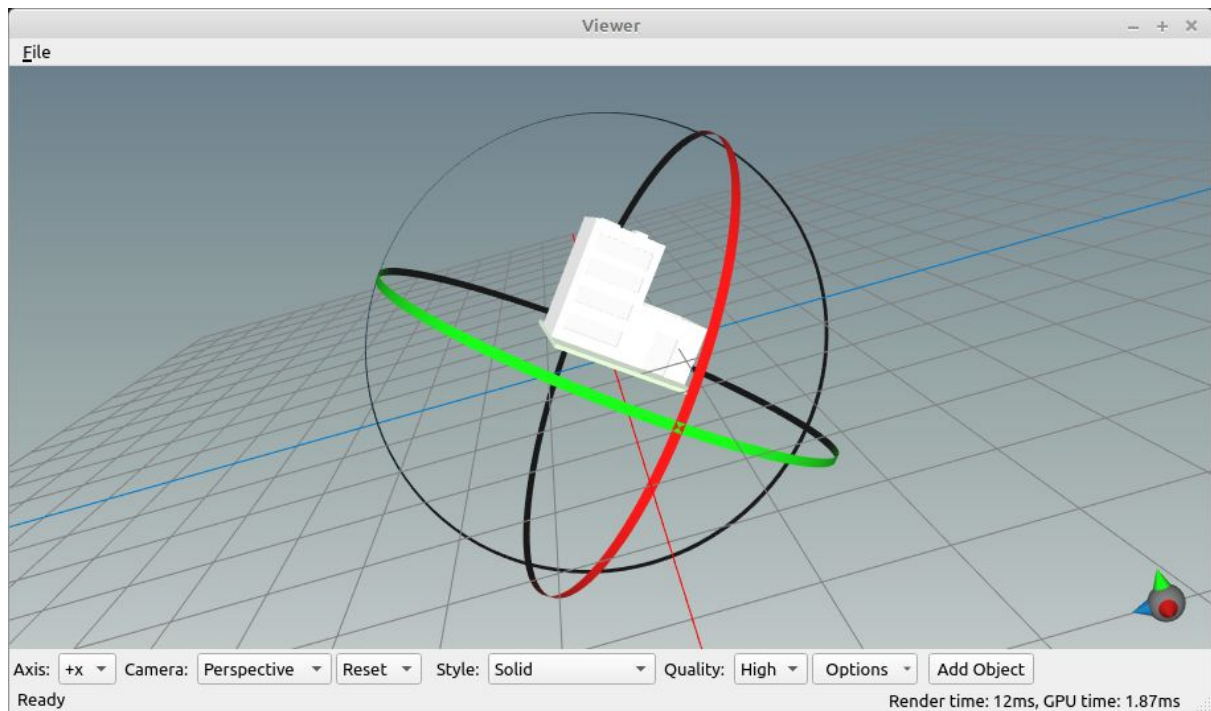
O manipulador da transformação é implementado pela classe **AxisMarker** (subclasse de **Group**), que é o conjunto de todas as geometrias possíveis (classes **Actor**). Esta classe é adicionado à cena como “ator do sistema” (não selecionável) e invisível. O **AxisMarker** recebe a matriz de transformação do objeto selecionado e cada manipulador recebe as informações necessárias desta matriz.

Quando uma transformação é selecionada (teclas **<T>**, **<R>** ou **<S>**), a classe **AxisMarker** chama o método **setVisible(True)** apenas para as partes que precisam estar visíveis. Quando um eixo é selecionado (teclas **<X>**, **<Y>** ou **<Z>**), é chamado o método **setHighlighted(True)** para as partes de devem estar realçadas.

O manipulador da translação é composta por 3 cones (classe de **Cone**) e 3 canos (classe **Pipe**, que é uma modificação da classe **Cylinder** sem suas extremidades) e seus eixos correspondem ao eixo da cena, pois cada classe recebe somente a posição da matriz de transformação do objeto.



O manipulador de rotação é composto por 3 canos (classe **Pipe**) simulando círculos. Cada classe recebe todas as informações da matriz de transformação, e portanto, a orientação dos eixos seguem a orientação do objeto.



O manipulador da escala é composta por 3 esferas (classe **Icosahedron**) e 3 canos (os mesmos da translação). Suas partes recebem todas as informações da matriz de transformação do objeto e portanto seus eixos refletem as transformações do objeto.

