

Computação Gráfica — Trabalho 1

Professor: Waldemar Celes
Aluno: Antenor Barros Leal

29 de setembro de 2024

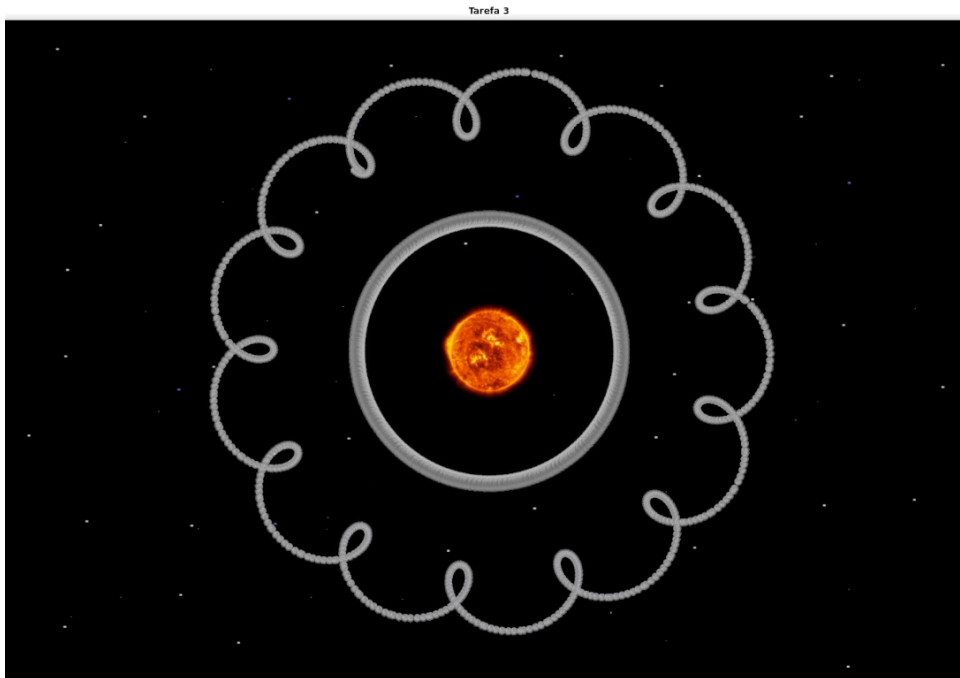
1 Resumo

Este trabalho tem como objetivo fazer uma representação simplificada do sistema solar, onde são mostrados o Sol, Mercúrio, a Terra e a Lua. A simulação usa transformações, texturas dentro de uma hierarquia de nodes, que organiza os movimentos de cena.

2 Movimentos compostos

Esta organização hierárquica permite ser possível movimentos compostos, como da Lua em torno da Terra e, ao mesmo tempo, do Sol.

Na imagem abaixo a Terra foi retirada e os frames da animação tiveram suas cores somadas. Note que o movimento em espiral da Lua é complexo se consideramos o movimento global.



3 Escalas e períodos

As escalas e distâncias dos planetas em relação ao Sol não são realistas, pois, para isso, os planetas teriam que ser muito pequenos e distantes um do outro, tornando-os difíceis de visualizar. Entretanto, as relações entre os períodos orbitais são realistas.

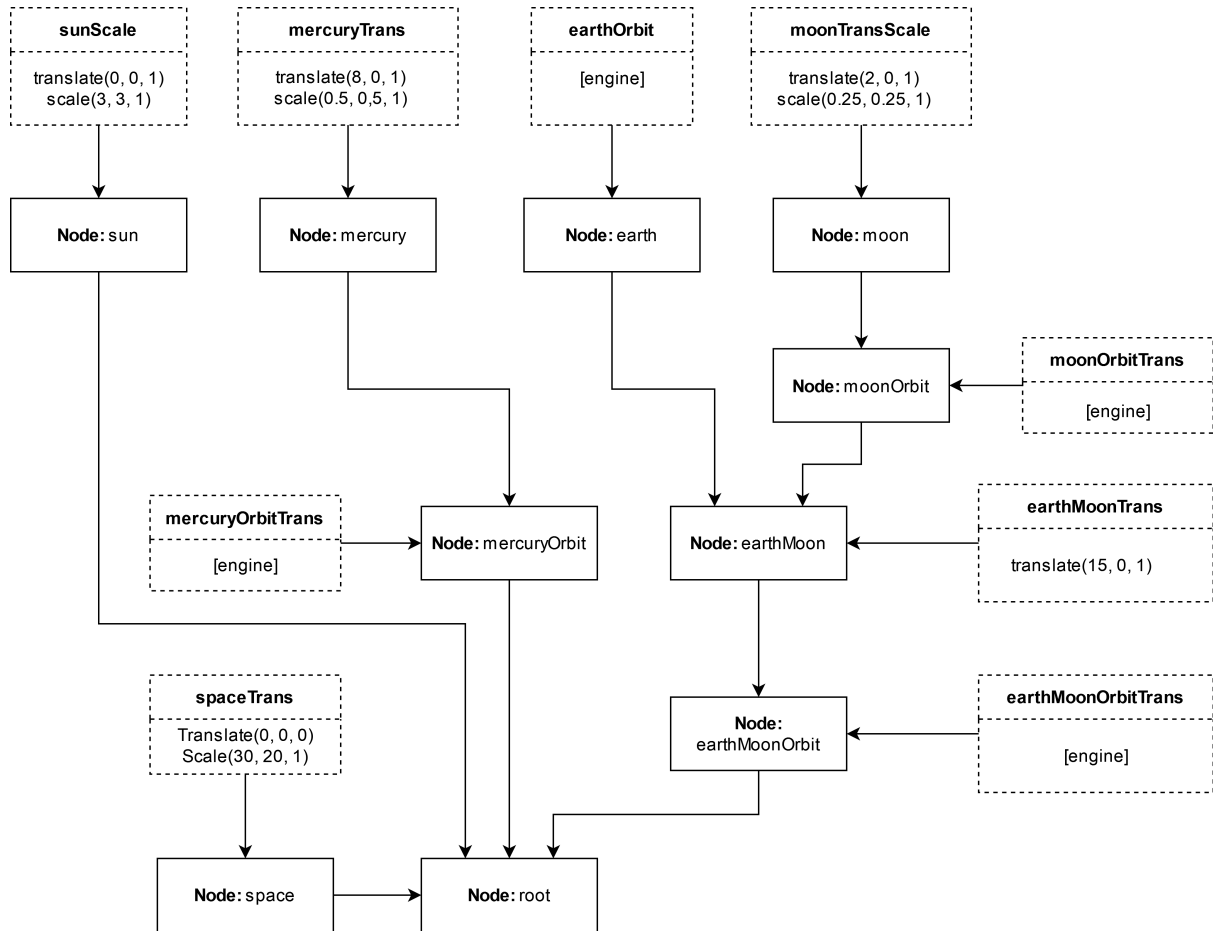
Na classe da engine, temos um valor padrão de 365 para o `m_update`, que é o período orbital da Terra em dias. Multiplicamos seu inverso por 5000 para que o movimento fique perceptível e multiplicamos por `dt` (o intervalo de tempo entre chamadas da função `update`) para que o movimento seja agnóstico em relação à velocidade do computador. A seguir, o código da função `update`.

```
virtual void Update (float dt)
{
    m_trf->Rotate(5000 / m_period * dt, 0, 0, 1);
}
```

Para a rotação da Terra (transformação `earthOrbit`), usei o valor 10. Embora isso não seja realista (o certo seria 1), foi necessário utilizar um valor mais alto para que o movimento de rotação da Terra não ficasse extremamente rápido.

4 Diagrama

O diagrama, a seguir, apresenta o grafo de cena, mostrando os objetos criados e suas conexões hierárquicas.



5 Nodes e Transformações

Os planetas e o Sol são representados por formas primitivas do tipo **Disk**, instanciadas nos nodes **sun**, **mercury**, **earth** e **moon**. Abaixo, o papel de cada node e suas respectivas transformações:

5.1 Sol

O node **sun** contém a transformação **sunScale**, que ajusta o tamanho e a posição do Sol. Esta transformação é usada para centralizar o Sol e mudar seu tamanho.

5.2 Mercúrio

O node **mercury** contém a transformação **mercuryTrans**, responsável por posicionar Mercúrio em sua órbita em torno do Sol. Ele é escalado para ser menor que a Terra e o Sol. O node **mercuryOrbit** envolve Mercúrio, servindo para rotacioná-lo em torno do Sol. Este node é controlado pelo **mercuryOrbitTrans**, que atualiza sua rotação com base no período da órbita de Mercúrio (88 dias, aproximadamente).

5.3 Terra e Lua

O node `earth` contém a transformação `earthOrbit`, que faz a Terra girar em torno dela mesma. Para a Lua, temos dois nodes de transformação:

- `moonTransScale`: responsável por posicionar a Lua em relação à Terra, ajustando sua escala e distância;
- `moonOrbit`: é um "container" que faz a Lua orbitar a Terra. Se aplicássemos o engine diretamente no moon, a Lua rotacionaria em torno de si mesma, em vez de orbitar a Terra.

O node `earthMoon` agrupa a Terra e a Lua, permitindo que ambas sejam posicionadas como um sistema em torno do Sol. Já o node `earthMoonOrbitTrans` é responsável pela rotação do sistema Terra-Lua em torno do Sol.

5.4 Espaço

O fundo da cena é representado por uma textura estrelada, aplicada a um quadrado escalado (primitiva `Square`). A transformação `spaceTrans` ajusta o tamanho e a posição desse quadrado para ocupar todo o plano de fundo. Para garantir que o fundo fique atrás dos planetas e do Sol, a componente z da transformação de escala é ajustada para um valor inferior, zero, no caso, enquanto todos os outros ficam com z igual a um.

6 Composição de Cena

A cena completa termina no nó root:

- O `root` contém os nodes do Sol, Mercúrio e o sistema Terra-Lua, além do fundo.
- As engines de movimentação que são responsáveis por atualizar as órbitas são ligadas usando o `scene->AddEngine`

7 Resultados

Essa composição hierárquica permite que a simulação funcione de maneira modular, onde cada corpo celeste tem suas transformações e atualizações associadas à sua posição local e movimento dentro do sistema solar.

Um vídeo de demonstração do correto funcionamento foi incluído: arquivo "demo.mp4".