

Universidade do Minho  
Licenciatura em Engenharia Informática

## Computação Gráfica

Fase 4 - Grupo 20

André Silva - A87958

Armando Silva - A87949

Joana Oliveira - A87956

João Nunes - A87972

Junho 2022



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Generator/Gerador</b>	<b>4</b>
2.1	Cálculos nas Primitivas . . . . .	4
2.1.1	Plano . . . . .	4
2.1.2	Caixa . . . . .	5
2.1.3	Cone . . . . .	5
2.1.4	Esfera . . . . .	6
2.1.5	<i>Teapot (Bezier Patches)</i> . . . . .	7
2.1.6	Toro . . . . .	7
2.1.7	Toro achatado ( <i>Flat Torus</i> ) . . . . .	8
2.2	Ficheiro .3d . . . . .	8
<b>3</b>	<b>Engine/Motor</b>	<b>9</b>
3.1	Leitura do Ficheiro XML . . . . .	9
3.2	Iluminação . . . . .	9
3.3	Texturas . . . . .	9
<b>4</b>	<b><i>Demos</i></b>	<b>10</b>
4.1	<i>Demos</i> de Algumas Primitivas . . . . .	10
4.2	Sistema Solar . . . . .	11
<b>5</b>	<b>Conclusão</b>	<b>12</b>
<b>A</b>	<b>Ficheiro XML</b>	<b>13</b>

## Lista de Figuras

1	Normal do plano . . . . .	4
2	Normais da caixa . . . . .	5
3	Normais do cone . . . . .	5
4	Normal da esfera . . . . .	6
5	Cálculo das tangentes de <i>Bezier</i> . . . . .	7
6	Normais do toro . . . . .	7
7	Ficheiro .3d . . . . .	8
8	<i>Demo 1</i> . . . . .	10
9	<i>Demo 2</i> . . . . .	10
10	<i>Demo 3</i> . . . . .	11
11	Sistema Solar . . . . .	11

# 1 Introdução

Nesta quarta e última fase do trabalho prático, é proposta a continuação do desenvolvimento do sistema solar num cenário gráfico 3D. Neste caso, foi pedida a adição das texturas em todos os elementos do sistema solar, e ainda, toda a iluminação.

Para o gerador, é necessária a alteração de cada primitiva para obtermos e escrevermos nos respetivos ficheiros 3d, os vetores normais e as coordenadas de textura.

Já para o motor, é inevitável a mudança da leitura dos ficheiros 3d e xml para conseguirmos lê-los com as novas alterações feitas. A aplicação das texturas e da iluminação também é uma implementação realizada no motor.

Neste relatório, apresentamos e explicamos todas as decisões tomadas que refletem na maneira como as funcionalidades pedidas foram implementadas. Tudo isto apresentado em várias secções.

## 2 Generator/Gerador

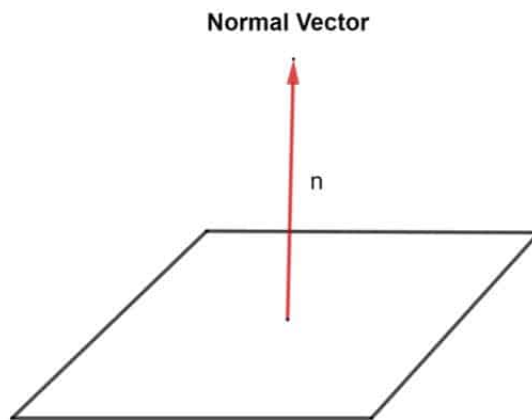
A primeira etapa desta fase passou no cálculo das coordenadas de textura e das normais de cada vértice de cada primitiva. Desta forma, conseguimos escrever no ficheiro 3d esses valores para a correta implementação das texturas e da iluminação por parte do Motor. Todas estas alterações foram realizadas nas respetivas funções no Gerador e são explicadas mais detalhadamente nesta secção.

### 2.1 Cálculos nas Primitivas

Apresentamos aqui a explicação para todos os cálculos efetuados para cada uma das primitivas, com o objetivo de obter as coordenadas da textura e as normais de cada vértice.

#### 2.1.1 Plano

O cálculo da normal de um plano é bastante simples. Uma vez que este se encontra no plano XZ, a normal dos pontos da face superior será  $(0,1,0)$  e da inferior  $(0,-1,0)$ .



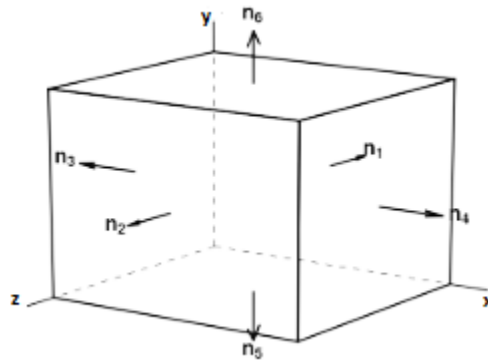
**Figura 1.** Normal do plano

O cálculo das texturas segue um raciocínio idêntico aquele usado para o desenho do plano. A textura é dividida utilizando o número de *divisions* dado e a cada iteração, pelas diversas divisões, o  $x$  aumenta em  $1/\text{divisions}$  até chegar à ponta contrária. De seguida, o  $y$  aumenta em  $1/\text{divisions}$ . Este processo é repetido até o plano estar completo.

### 2.1.2 Caixa

O cálculo das **normais** da caixa é também bastante simples, pois as normais de cada face não necessitam de cálculos. Assumindo que as variáveis presentes na figura a seguir representam os vetores normais de cada face, podemos concluir que:

- O vetor  $n_1$  é  $(0,0,-1)$ .
- O vetor  $n_2$  é  $(0,0,1)$ .
- O vetor  $n_3$  é  $(-1,0,0)$ .
- O vetor  $n_4$  é  $(1,0,0)$ .
- O vetor  $n_5$  é  $(0,-1,0)$ .
- O vetor  $n_6$  é  $(0,1,0)$ .



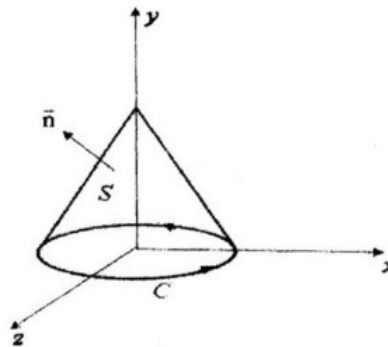
**Figura 2.** Normais da caixa

Para o cálculo das **texturas**, segue-se o mesmo método usado nas texturas do plano para cada face.

### 2.1.3 Cone

No cálculo das **normais** da base do cone, visto que este está contido no plano XZ, o seu vetor normal será  $(0,-1,0)$ . Já na superfície lateral do cone, o valor da normal será:

$$\vec{n} = (\sin(\alpha), \cos(\text{atan}(\text{height}/\text{radius})), \cos(\alpha))$$



**Figura 3.** Normais do cone

Relativamente às **texturas**, o cálculo é feito da seguinte forma:

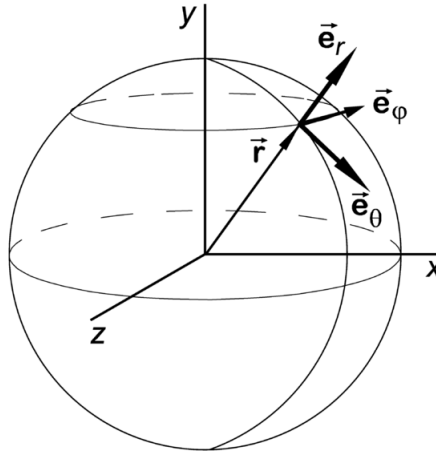
- Para a base, como sabemos que o comprimento da base de cada triângulo é igual ao perímetro da circunferência a dividir pelas *slices*, então sabemos que o comprimento é  $var = 2 * PI * Raio / slices$ . Sendo assim, cada coordenada da textura poderá ter os seguintes valores  $(i * var, 0, 0)$ ,  $((i+1) * var, 0, 0)$  e  $(var/2, 1, 0)$ , sendo que  $i$  representa a iteração entre as *slices*.
- Para a superfície lateral, a cada iteração das *slices* a variável  $X$  da textura incrementa  $1/slices$  e depois volta a 0, e a cada iteração das *stacks* a variável  $Y$  da textura aumenta  $1/stacks$ , as coordenadas terão um valor entre  $(j * (1/slices), i * (1/stacks))$ , onde  $i$  e  $j$  representam o número da iteração das *slices* e *stacks* respetivamente.

#### 2.1.4 Esfera

As **normais** da esfera são facilmente calculadas, pois apenas é necessário os ângulos presentes para o cálculo das vértices, isto é, as normais têm a seguinte forma:

$$\vec{n} = (\cos(\beta) * \sin(\alpha), \sin(\beta), \cos(\beta) * \cos(\alpha))$$

Onde  $\beta$  representa a variação das *stacks* e  $\alpha$  a variação das *slices*.



**Figura 4.** Normal da esfera

Nas **texturas** o raciocínio utilizado é semelhante ao do cone.

### 2.1.5 Teapot (Bezier Patches)

Para o cálculo das **normais**, é necessário calcular as seguintes tangentes:

$$\frac{\partial B(u,v)}{\partial u} = [3u^2 \quad 2u \quad 1 \quad 0] M \begin{bmatrix} P_{00} & P_{01} & P_{02} & P_{03} \\ P_{10} & P_{11} & P_{12} & P_{13} \\ P_{20} & P_{21} & P_{22} & P_{23} \\ P_{30} & P_{31} & P_{32} & P_{33} \end{bmatrix} M^T V^T$$

$$\frac{\partial B(u,v)}{\partial v} = U M \begin{bmatrix} P_{00} & P_{01} & P_{02} & P_{03} \\ P_{10} & P_{11} & P_{12} & P_{13} \\ P_{20} & P_{21} & P_{22} & P_{23} \\ P_{30} & P_{31} & P_{32} & P_{33} \end{bmatrix} M^T \begin{bmatrix} 3v^2 \\ 2v \\ 1 \\ 0 \end{bmatrix}$$

**Figura 5.** Cálculo das tangentes de *Bezier*

A normal do ponto será o resultado normalizado da multiplicação das tangentes calculadas.

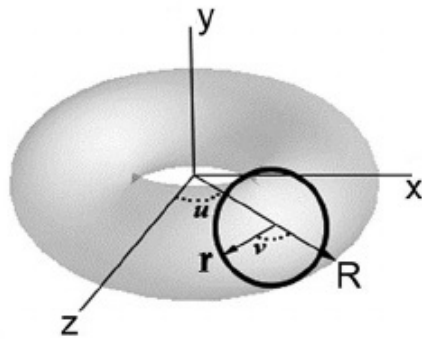
Já nas **texturas** as coordenadas dos pontos de textura são definidas por (1-v,1-u), onde v e u são valores de tecelagem.

### 2.1.6 Toro

A **normal** do Toro de um ponto é obtido da seguinte forma:

$$\vec{r} = (\cos(\alpha) * \cos(\beta), \sin(\alpha), \cos(\alpha) * \sin(\beta))$$

Onde  $\beta$  representa a variação das *stacks* e  $\alpha$  a variação das *slices*.



**Figura 6.** Normais do toro

A **textura** também é semelhante à da esfera.



### 2.1.7 Toro achatado (*Flat Torus*)

As **normais** desta primitiva são equivalente às do plano, isto é, a face superior é (0,1,0) e a face inferior é (0,-1,0).

Para a **textura** é necessário dividir o y em *slices*, onde a cada iteração das *slices* o x varia entre 0 e 1 e o y aumenta ( $1/slices$ ).

## 2.2 Ficheiro .3d

Com a adição das normais e das texturas, o ficheiro .3d passa agora a ter três tipos de pontos diferentes. A representação dos pontos no ficheiro é a que se encontra apresentada na figura em baixo, com um ponto, a normal do ponto e a textura do ponto. Repetindo depois para o total de pontos, que se encontra representado no início do ficheiro.

```
324
-1.000000,-1.000000,-1.000000
0,-1,0
0.000000,0.000000,0
-0.333333,-1.000000,-0.333333
0,-1,0
0.333333,0.333333,0
-1.000000,-1.000000,-0.333333
0,-1,0
0.000000,0.333333,0
-1.000000,-1.000000,-1.000000
0,-1,0
0.000000,0.000000,0
```

Figura 7. Ficheiro .3d

## 3 Engine/Motor

Em relação ao Motor foi necessária a modificação da leitura dos ficheiros XML, de forma a suportar as novas alterações realizadas no Gerador. É ainda preciso efetuar algumas mudanças em algumas funções para conseguirmos aplicar corretamente as texturas e a iluminação. Tudo isto vai ser explicado pormenorizadamente nesta secção.

### 3.1 Leitura do Ficheiro XML

A nova informação sobre luzes e texturas encontra-se representada no ficheiro XML, isto força uma atualização da função que realiza a leitura do mesmo. Esta função é agora capaz de reconhecer texturas e todos os tipos de luz e as suas propriedades. As propriedades da luz são guardadas num *array* global *all\_lights* enquanto que os ficheiros de textura são guardados no seu respetivo *group* para sabermos a que figura pertencem.

### 3.2 Iluminação

Neste momento é possível ter vários tipos de iluminação, como a iluminação global, e a iluminação específica a cada figura. Deste modo, e como mencionado acima, a luz tem várias propriedades, sendo que a luz global admite **três tipos**: posicional, direcional ou foco. Sendo o primeiro caracterizado apenas pela sua posição, o segundo pela direção da luz e o terceiro pela posição e direção da luz.

Já a luz referente a cada figura pode conter **um ou mais dos seguintes elementos**: difusa, especular, ambiente e emissiva. Além disso, pode também ter *shininess*.

Como o programa pode conter vários tipos de iluminação em simultâneo foi necessário implementar o comando *glEnable(GL\_LIGHT0 + i)*, em que *i* é um contador que é incrementado até ao número total de luzes, de modo a ativar todas.

O grupo achou por bem definir um valor *default* para as luzes de cada figura, sendo estes valores de (200,200,200) para a luz difusa, (50,50,50) para a luz especular e uma *shininess* de 0.

### 3.3 Texturas

Para esta última fase foi necessário fazer a aplicação de textura aos diversos objetos. Isto foi conseguido começando por carregar as texturas para memória gráfica usando a função *loadTexture*. Esta função recebe como input o *path* para o ficheiro da textura.

Tal como para a luz, é necessário ativar o uso de texturas através das funções *glEnable(GL\_TEXTURE\_2D)* e *glEnableClientState(GL\_TEXTURE\_COORD\_ARRAY)*.

## 4 *Demos*

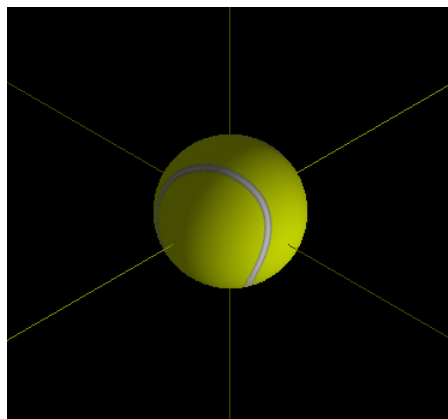
Para finalizar, decidimos apresentar, nesta secção, algumas *demos* com primitivas individuais que mostram a aplicação das texturas e da iluminação. Exibimos ainda, o sistema solar final obtido com todas as funcionalidades pedidas desde a primeira fase até esta última.

### 4.1 *Demos* de Algumas Primitivas

Decidimos apresentar umas simples *demos* que combinam algumas primitivas com textura e iluminação.

#### *Demo 1*

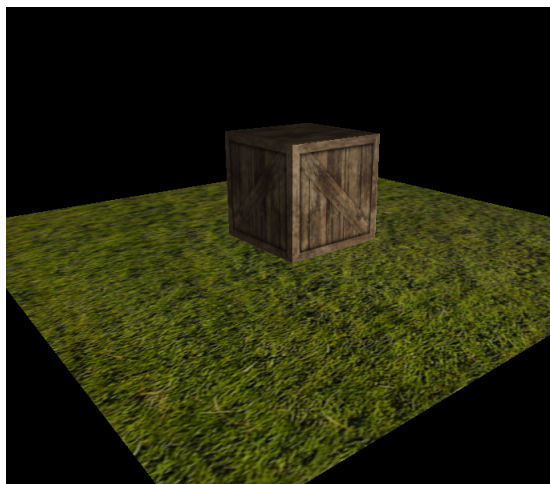
Uma simples esfera, com textura e iluminação.



**Figura 8.** *Demo 1*

#### *Demo 2*

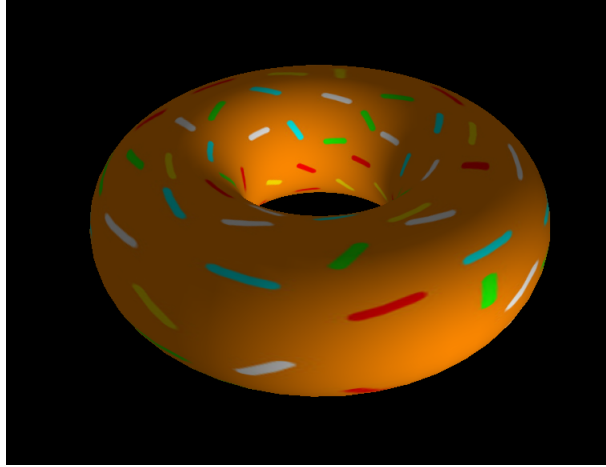
Uma caixa e um plano, com textura e iluminação.



**Figura 9.** *Demo 2*

### ***Demo 3***

Um toro, com textura e iluminação.



**Figura 10.** *Demo 3*

## **4.2 Sistema Solar**

Apresentamos agora o sistema solar final, com tudo o que fomos aprendendo e aplicando em todas as fases realizadas. Nesta última adicionamos a iluminação e as texturas. De salientar, que adicionamos uma textura para o fundo do sistema solar, tornando-o ainda mais realista, e que toda a luz é proveniente do sol. Como extra, adicionamos uma textura de fundo do espaço de forma a tornar o sistema solar mais realista, esta adição pode ser ativada e desativada clicando na tecla 'X'.



**Figura 11.** Sistema Solar

## 5 Conclusão

Dado por concluída a quarta e última fase deste trabalho prático, consideramos importante realçar todos os pontos positivos e negativos, e ainda, efetuar uma análise crítica final do trabalho realizado.

Todos os requisitos estarem funcionais é um ponto bastante positivo. Acrescentamos ainda uma textura de fundo que aproxima o sistema solar da realidade. As iluminação funciona para qualquer número de luzes, logo que estejam dentro do máximo permitido de 7.

O grupo acha que o projeto poderia estar mais completo se tivessemos adicionado a câmara fps, infelizmente tivemos algumas dificuldades na implementação das luzes o que não restou tempo suficiente para a implementação da câmara.

Para concluir, o grupo sente que o trabalho realizado nesta fase cumpre com todos os requisitos propostos, e como tal, considera o trabalho positivo. Acrescentamos ainda, que o resultado final obtido é satisfatório e representa bem o empenho e dedicação do grupo.

## A Ficheiro XML

```
<world>
  <camera>
    <position x="10" y="10" z="10"/>
    <lookAt x="0" y="0" z="0"/>
    <up x="0" y="1" z="0"/>
    <projection fov="60" near="1" far="1000"/>
  </camera>
  <group>
    <!--SOL-->
    <transform>
      <color r="0.98" g="0.73" b="0.01"/>
      <rotate time="3" x="0" y="1" z="0" />
    </transform>
    <models>
      <model file="sphere.3d"/>
    </models>
  </group>
  <group>
    <!--MERCURIO-->
    <transform>
      <color r="0.5" g="0.31" b="0.0"/>
      <translate time = "3" align="True">
        <point x = "1.41421" y = "0" z = "1.41421" />
        <point x = "0" y = "0" z = "2" />
        <point x = "-1.41421" y = "0" z = "1.41421" />
        <point x = "-2" y = "0" z = "0" />
        <point x = "-1.41421" y = "0" z = "-1.41421" />
        <point x = "0" y = "0" z = "-2" />
        <point x = "1.41421" y = "0" z = "-1.41421" />
        <point x = "2" y = "0" z = "0" />
      </translate>
      <rotate time="5" x="0" y="1" z="0" />
      <rotate angle="-30.0" x="0.0" y="1.0" z="0.0"/>
      <scale x="0.1" y="0.1" z="0.1"/>
    </transform>
    <models>
      <model file="sphere.3d"/>
    </models>
  </group>
  <group>
    <!--VENUS-->
    <transform>
      <color r="0.82" g="0.27" b="0.0"/>
      <translate time = "7" align="True">
        <point x = "3.5" y = "0" z = "0" />
        <point x = "2.47487" y = "0" z = "-2.47487" />
        <point x = "0" y = "0" z = "-3.5" />
        <point x = "-2.47487" y = "0" z = "-2.47487" />
        <point x = "-3.5" y = "0" z = "0" />
        <point x = "-2.47487" y = "0" z = "2.47487" />
        <point x = "0" y = "0" z = "3.5" />
      </translate>
    </transform>
  </group>
</world>
```

```

        <point x = "2.47487" y = "0" z = "2.47487" />
    </translate>
    <rotate time="7" x="0" y="-1" z="0" />
    <rotate angle="90" x="0" y="1" z="0"/>
    <scale x="0.2" y="0.2" z="0.2"/>
</transform>
<models>
    <model file="sphere.3d"/>
</models>
</group>
<group>
    <!--TERRA-->
    <transform>
        <color r="0.0" g="0.59" b="0.86"/>
        <translate time = "10" align="True">
            <point x = "4.5" y = "0" z = "0" />
            <point x = "3.18198" y = "0" z = "-3.18198" />
            <point x = "0" y = "0" z = "-4.5" />
            <point x = "-3.18198" y = "0" z = "-3.18198" />
            <point x = "-4.5" y = "0" z = "0" />
            <point x = "-3.18198" y = "0" z = "3.18198" />
            <point x = "0" y = "0" z = "4.5" />
            <point x = "3.18198" y = "0" z = "3.18198" />
        </translate>
        <rotate time="3" x="0" y="1" z="0" />
        <rotate angle="233" x="0" y="1" z="0"/>
        <scale x="0.3" y="0.3" z="0.3"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
    <group>
        <!--LUA-->
        <transform>
            <color r="1.0" g="1.0" b="1.0"/>
            <rotate angle="26.57" x="1" y="0.0" z="0.0"/>
            <translate time = "10" align="True">
                <point x = "2.236068" y = "0" z = "0" />
                <point x = "1.58113" y = "0" z = "-1.58113" />
                <point x = "0" y = "0" z = "-2.236068" />
                <point x = "-1.58113" y = "0" z = "-1.58113" />
                <point x = "-2.236068" y = "0" z = "0" />
                <point x = "-1.58113" y = "0" z = "1.58113" />
                <point x = "0" y = "0" z = "2.236068" />
                <point x = "1.58113" y = "0" z = "1.58113" />
            </translate>
            <scale x="0.25" y="0.25" z="0.25"/>
        </transform>
        <models>
            <model file="sphere.3d"/>
        </models>
    </group>
</group>
<group>

```

```

<!--MARTE-->
<transform>
  <color r="0.85" g="0.46" b="0.007"/>
  <translate time = "20" align="True">
    <point x = "6" y = "0" z = "0" />
    <point x = "4.24264" y = "0" z = "-4.24264" />
    <point x = "0" y = "0" z = "-6" />
    <point x = "-4.24264" y = "0" z = "-4.24264" />
    <point x = "-6" y = "0" z = "0" />
    <point x = "-4.24264" y = "0" z = "4.24264" />
    <point x = "0" y = "0" z = "6" />
    <point x = "4.24264" y = "0" z = "4.24264" />
  </translate>
  <rotate time="3" x="0" y="1" z="0" />
  <rotate angle="260" x="0" y="1" z="0"/>
  <scale x="0.15" y="0.15" z="0.15"/>
</transform>
<models>
  <model file="sphere.3d"/>
</models>
<group>
  <!--LUA1-->
  <transform>
    <color r="1.0" g="1.0" b="1.0"/>
    <rotate angle="-38.66" x="1" y="0.0" z="0.0"/>
    <translate time = "10" align="True">
      <point x = "1.600781" y = "0" z = "0" />
      <point x = "1.13192" y = "0" z = "-1.13192" />
      <point x = "0" y = "0" z = "-1.600781" />
      <point x = "-1.13192" y = "0" z = "-1.13192" />
      <point x = "-1.600781" y = "0" z = "0" />
      <point x = "-1.13192" y = "0" z = "1.13192" />
      <point x = "0" y = "0" z = "1.600781" />
      <point x = "1.13192" y = "0" z = "1.13192" />
    </translate>
    <scale x="0.25" y="0.25" z="0.25"/>
  </transform>
  <models>
    <model file="sphere.3d"/>
  </models>
</group>
<group>
  <!--LUA2-->
  <transform>
    <color r="0.82" g="0.73" b="0.55"/>
    <rotate angle="33.69" x="1" y="0.0" z="0.0"/>
    <translate time = "10" align="True">
      <point x = "1.802776" y = "0" z = "0" />
      <point x = "1.27475" y = "0" z = "-1.27475" />
      <point x = "0" y = "0" z = "-1.802776" />
      <point x = "-1.27475" y = "0" z = "-1.27475" />
      <point x = "-1.802776" y = "0" z = "0" />
      <point x = "-1.1.27475" y = "0" z = "1.27475" />
      <point x = "0" y = "0" z = "1.802776" />
    </translate>
  </transform>
</group>

```



```

        <point x = "1.27475" y = "0" z = "1.27475" />
    </translate>
    <scale x="0.3" y="0.3" z="0.3"/>
</transform>
<models>
    <model file="sphere.3d"/>
</models>
</group>
</group>
<group>
    <!--JUPITER-->
    <transform>
        <color r="1.0" g="0.74" b="0.48"/>
        <translate time = "30" align="True">
            <point x = "8" y = "0" z = "0" />
            <point x = "5.65685" y = "0" z = "-5.65685" />
            <point x = "0" y = "0" z = "-8" />
            <point x = "-5.65685" y = "0" z = "-5.65685" />
            <point x = "-8" y = "0" z = "0" />
            <point x = "-5.65685" y = "0" z = "5.65685" />
            <point x = "0" y = "0" z = "8" />
            <point x = "5.65685" y = "0" z = "5.65685" />
        </translate>
        <rotate time="6" x="0" y="1" z="0" />
        <rotate angle="150" x="0" y="1" z="0"/>
        <scale x="0.7" y="0.7" z="0.7"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
    <group>
        <!--LUA1-->
        <transform>
            <color r="0.82" g="0.27" b="0.0"/>
            <rotate angle="0.1" x="1" y="0.0" z="0.0"/>
            <translate time = "10" align="True">
                <point x = "2" y = "0" z = "0" />
                <point x = "1.41421" y = "0" z = "-1.41421" />
                <point x = "0" y = "0" z = "-2" />
                <point x = "-1.41421" y = "0" z = "-1.41421" />
                <point x = "-2" y = "0" z = "0" />
                <point x = "-1.41421" y = "0" z = "1.41421" />
                <point x = "0" y = "0" z = "2" />
                <point x = "1.41421" y = "0" z = "1.41421" />
            </translate>
            <scale x="0.1" y="0.1" z="0.1"/>
        </transform>
        <models>
            <model file="sphere.3d"/>
        </models>
    </group>
</group>
    <!--LUA2-->
    <transform>

```

```

    <color r="1.0" g="1.0" b="1.0"/>
    <rotate angle="-25" x="1" y="0.0" z="0.0"/>
    <translate time = "10" align="True">
        <point x = "1.732051" y = "0" z = "0" />
        <point x = "1.22474" y = "0" z = "-1.22474" />
        <point x = "0" y = "0" z = "-1.732051" />
        <point x = "-1.22474" y = "0" z = "-1.22474" />
        <point x = "-1.732051" y = "0" z = "0" />
        <point x = "-1.22474" y = "0" z = "1.22474" />
        <point x = "0" y = "0" z = "1.732051" />
        <point x = "1.22474" y = "0" z = "1.22474" />
    </translate>
    <scale x="0.1" y="0.1" z="0.1"/>
</transform>
<models>
    <model file="sphere.3d"/>
</models>
</group>
<group>
    <!--LUA3-->
    <transform>
        <color r="0.92" g="0.8" b="0.38"/>
        <rotate angle="35.26" x="1" y="0.0" z="0.0"/>
        <translate time = "10" align="True">
            <point x = "1.732051" y = "0" z = "0" />
            <point x = "1.22474" y = "0" z = "-1.22474" />
            <point x = "0" y = "0" z = "-1.732051" />
            <point x = "-1.22474" y = "0" z = "-1.22474" />
            <point x = "-1.732051" y = "0" z = "0" />
            <point x = "-1.22474" y = "0" z = "1.22474" />
            <point x = "0" y = "0" z = "1.732051" />
            <point x = "1.22474" y = "0" z = "1.22474" />
        </translate>
        <scale x="0.15" y="0.15" z="0.15"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>
<group>
    <!--LUA4-->
    <transform>
        <color r="0.55" g="0.42" b="0.26"/>
        <rotate angle="-35.26" x="1" y="0.0" z="0.0"/>
        <translate time = "10" align="True">
            <point x = "1.732051" y = "0" z = "0" />
            <point x = "1.22474" y = "0" z = "-1.22474" />
            <point x = "0" y = "0" z = "-1.732051" />
            <point x = "-1.22474" y = "0" z = "-1.22474" />
            <point x = "-1.732051" y = "0" z = "0" />
            <point x = "-1.22474" y = "0" z = "1.22474" />
            <point x = "0" y = "0" z = "1.732051" />
            <point x = "1.22474" y = "0" z = "1.22474" />
        </translate>

```

```

        <scale x="0.15" y="0.15" z="0.15"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>
</group>
<group>
    <!--SATURNO-->
    <transform>
        <color r="0.56" g="0.83" b="0.42"/>
        <translate time = "40" align="True">
            <point x = "9.5" y = "0" z = "0" />
            <point x = "6.71751" y = "0" z = "-6.71751" />
            <point x = "0" y = "0" z = "-9.5" />
            <point x = "-6.71751" y = "0" z = "-6.71751" />
            <point x = "-9.5" y = "0" z = "0" />
            <point x = "-6.71751" y = "0" z = "6.71751" />
            <point x = "0" y = "0" z = "9.5" />
            <point x = "6.71751" y = "0" z = "6.71751" />
        </translate>
        <rotate time="5" x="0" y="1" z="0" />
        <rotate angle="-50" x="0" y="1" z="0"/>
        <scale x="0.65" y="0.65" z="0.65"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
    <group>
        <!--ANEL-->
        <transform>
            <rotate angle="-30" x="1" y="0.0" z="0.0"/>
        </transform>
        <models>
            <model file="torus.3d"/>
        </models>
    </group>
    <group>
        <!--LUA1-->
        <transform>
            <color r="0.48" g="0.63" b="0.38"/>
            <rotate angle="45" x="1" y="0.0" z="0.0"/>
            <translate time = "10" align="True">
                <point x = "1.414214" y = "0" z = "0" />
                <point x = "1" y = "0" z = "-1" />
                <point x = "0" y = "0" z = "-1.414214" />
                <point x = "-1" y = "0" z = "-1" />
                <point x = "-1.414214" y = "0" z = "0" />
                <point x = "-1" y = "0" z = "1" />
                <point x = "0" y = "0" z = "1.414214" />
                <point x = "1" y = "0" z = "1" />
            </translate>
            <scale x="0.15" y="0.15" z="0.15"/>
        </transform>
    </group>

```

```

        <models>
            <model file="sphere.3d"/>
        </models>
    </group>
    <group>
        <!--LUA2-->
        <transform>
            <color r="1.0" g="1.0" b="1.0"/>
            <rotate angle="33.69" x="1" y="0.0" z="0.0"/>
            <translate time = "10" align="True">
                <point x = "1.802776" y = "0" z = "0" />
                <point x = "1.27475" y = "0" z = "-1.27475" />
                <point x = "0" y = "0" z = "-1.802776" />
                <point x = "-1.27475" y = "0" z = "-1.27475" />
                <point x = "-1.802776" y = "0" z = "0" />
                <point x = "-1.27475" y = "0" z = "1.27475" />
                <point x = "0" y = "0" z = "1.802776" />
                <point x = "1.27475" y = "0" z = "1.27475" />
            </translate>
            <scale x="0.1" y="0.1" z="0.1"/>
        </transform>
        <models>
            <model file="sphere.3d"/>
        </models>
    </group>
    <group>
        <!--LUA3-->
        <transform>
            <color r="0.5" g="0.31" b="0.0"/>
            <rotate angle="-56.31" x="1" y="0.0" z="0.0"/>
            <translate time = "10" align="True">
                <point x = "1.802776" y = "0" z = "0" />
                <point x = "1.27475" y = "0" z = "-1.27475" />
                <point x = "0" y = "0" z = "-1.802776" />
                <point x = "-1.27475" y = "0" z = "-1.27475" />
                <point x = "-1.802776" y = "0" z = "0" />
                <point x = "-1.27475" y = "0" z = "1.27475" />
                <point x = "0" y = "0" z = "1.802776" />
                <point x = "1.27475" y = "0" z = "1.27475" />
            </translate>
            <scale x="0.15" y="0.15" z="0.15"/>
        </transform>
        <models>
            <model file="sphere.3d"/>
        </models>
    </group>
</group>
<group>
    <!--URANO-->
    <transform>
        <color r="0.35" g="0.75" b="0.86"/>
        <translate time = "50" align="True">
            <point x = "10.5" y = "0" z = "0" />
            <point x = "7.42462" y = "0" z = "-7.42462" />

```

```

        <point x = "0" y = "0" z = "-10.5" />
        <point x = "-7.42462" y = "0" z = "-7.42462" />
        <point x = "-10.5" y = "0" z = "0" />
        <point x = "-7.42462" y = "0" z = "7.42462" />
        <point x = "0" y = "0" z = "10.5" />
        <point x = "7.42462" y = "0" z = "7.42462" />
    </translate>
    <rotate time="2.5" x="0" y="-1" z="0"/>
    <rotate angle="39" x="0" y="1" z="0"/>
    <scale x="0.25" y="0.25" z="0.25"/>
</transform>
<models>
    <model file="sphere.3d"/>
</models>
<group>
    <!--LUA1-->
    <transform>
        <color r="1.0" g="1.0" b="1.0"/>
        <rotate angle="29.02" x="1" y="0.0" z="0.0"/>
        <translate time = "10" align="True">
            <point x = "2.061553" y = "0" z = "0" />
            <point x = "1.45773" y = "0" z = "-1.45773" />
            <point x = "0" y = "0" z = "-2.061553" />
            <point x = "-1.45773" y = "0" z = "-1.45773" />
            <point x = "-2.061553" y = "0" z = "0" />
            <point x = "-1.45773" y = "0" z = "1.45773" />
            <point x = "0" y = "0" z = "2.061553" />
            <point x = "1.45773" y = "0" z = "1.45773" />
        </translate>
        <scale x="0.15" y="0.15" z="0.15"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>
<group>
    <!--LUA2-->
    <transform>
        <color r="0.87" g="0.59" b="0.28"/>
        <rotate angle="56.31" x="1" y="0.0" z="0.0"/>
        <translate time = "10" align="True">
            <point x = "1.802776" y = "0" z = "0" />
            <point x = "1.27475" y = "0" z = "-1.27475" />
            <point x = "0" y = "0" z = "-1.802776" />
            <point x = "-1.27475" y = "0" z = "-1.27475" />
            <point x = "-1.802776" y = "0" z = "0" />
            <point x = "-1.27475" y = "0" z = "1.27475" />
            <point x = "0" y = "0" z = "1.802776" />
            <point x = "1.27475" y = "0" z = "1.27475" />
        </translate>
        <scale x="0.25" y="0.25" z="0.25"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>

```

```

        </models>
    </group>
</group>
<group>
    <!--NEPTUNO-->
    <transform>
        <color r="0.11" g="0.33" b="0.58"/>
        <translate time = "100" align="True">
            <point x = "12" y = "0" z = "0" />
            <point x = "8.48528" y = "0" z = "-8.48528" />
            <point x = "0" y = "0" z = "-12" />
            <point x = "-8.48528" y = "0" z = "-8.48528" />
            <point x = "-12" y = "0" z = "0" />
            <point x = "-8.48528" y = "0" z = "8.48528" />
            <point x = "0" y = "0" z = "12" />
            <point x = "8.48528" y = "0" z = "8.48528" />
        </translate>
        <rotate time="2" x="0" y="1" z="0" />
        <rotate angle="15" x="0" y="1" z="0"/>
        <scale x="0.2" y="0.2" z="0.2"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>
<!--LUA-->
    <transform>
        <color r="0.98" g="0.73" b="0.01"/>
        <rotate angle="-26.57" x="1" y="0.0" z="0.0"/>
        <translate time = "10" align="True">
            <point x = "2.236068" y = "0" z = "0" />
            <point x = "1.58113" y = "0" z = "-1.58113" />
            <point x = "0" y = "0" z = "-2.236068" />
            <point x = "-1.58113" y = "0" z = "-1.58113" />
            <point x = "-2.236068" y = "0" z = "0" />
            <point x = "-1.58113" y = "0" z = "1.58113" />
            <point x = "0" y = "0" z = "2.236068" />
            <point x = "1.58113" y = "0" z = "1.58113" />
        </translate>
        <scale x="0.25" y="0.25" z="0.25"/>
    </transform>
    <models>
        <model file="sphere.3d"/>
    </models>
</group>
</group>
<group>
    <!--COMETA-->
    <transform>
        <color r="0.5" g="0.31" b="0.0"/>
        <translate time = "20" align="True">
            <point x = "-1" y = "-3" z = "-4" />
            <point x = "-3" y = "0" z = "4" />
            <point x = "4" y = "3" z = "2" />

```

```
        <point x = "3" y = "0" z = "0" />
    </translate>
    <scale x="0.1" y="0.1" z="0.1"/>
</transform>
<models>
    <model file="bezier.3d"/>
</models>
</group>
</world>
```