

INSTITUTO TECNOLÓGICO DE AERONÁUTICA

Fundamentos de Computação Gráfica

CCI-36



Laboratório 1

COMP 20

Heládio Sampaio Lopes

Sebastião Beethoven Brandão Filho

Professor:

Carlos Henrique Q. Forster

ITA-SJC-2019



Descrição do código implementado

O Laboratório proposto possui como objetivo a criação de uma cena em 3D utilizando a biblioteca THREE.JS. Para a sua realização, foram seguidos determinados pré-requisitos estabelecidos.

Os trechos de código descritos a seguir promovem a validação dos pré-requisitos para cada uma das tarefas propostas.

1. Formar objetos por primitivas geométricas

Para criar os objetos da cena foram usados primitivas geométricas tais como paralelepípedos (BoxGeometry), cilindros (CylinderGeometry), e círculos (CircleGeometry).

O trecho de código abaixo ilustra o uso de paralelepípedos para criação da mesa.

```
1 // FUNCAO MESA
2 function makeTable(leg_col=0xff5522, cover_col=0x22cc22) {
3     var leg_mat = new THREE.MeshLambertMaterial({color:leg_col});
4     var cover_mat = new THREE.MeshLambertMaterial({color:cover_col});
5
6     // Pernas
7     var leg_geom = new THREE.BoxGeometry(0.5,3.0,0.5)
8     var leg = new THREE.Mesh(leg_geom, leg_mat)
9
10    (...)
11
12    var cover_geom = new THREE.BoxGeometry(10,0.3,10);
13    var cover=new THREE.Mesh(cover_geom, cover_mat);
14    cover.position.set(0,3.0,0);
15
16    (...)
17
18 }
```

Listagem 1: Fragmento de código da função makeTable().



2. Realizar transformações geométricas posicionando/alterando objetos

Para relizar as transformações geométricas foi definido uma variável tempo, e assim, alterar o posicionamento dos objetos conforme o passar do tempo.

O fragmento de código abaixo ilustra a alteração do posicionamento do objeto prato com o tempo.

```
1 var t=0
2 var animate = function () {
3     requestAnimationFrame( animate );    controls.update();
4
5     // Movimento
6     t = (t+1) % 60;
7
8     g_plate.rotation.y = t*Math.PI/60.0;
9
10    if (t < 30) {
11        g_plate.position.y = t/10;
12    } else {
13        g_plate.position.y = 6 - t/10;
14    }
15
16    renderer.render( scene, camera );
17 };
18
19 animate();
```

Listagem 2: Fragmento do código que ilustra transformações dos objetos.



3. Organizar hierarquicamente os objetos através de grupos

Para realizar a organização hierárquica foram criados alguns objetos (seat, backrest) e um grupo de objetos (g_leg). Posteriormente foi criado um novo grupo majoritário que organiza todos os componentes descritos, formando o objeto cadeira em si.

O fragmento de código abaixo ilustra a organização hierárquica de objetos através de grupos.

```
1 // FUNCAO CADEIRA
2 function makeChair(col=0x808080) {
3     (...)
4
5     // Agrupando
6     var g_chair = new THREE.Group();
7     g_chair.add(g_leg);
8     g_chair.add(seat);
9     g_chair.add(backrest);
10    g_chair.position.y = 1;
11
12    return g_chair;
13 }
```

Listagem 3: Fragmento do código da função makeChair().

4. Construir objetos por meio de subrotinas

As Listagens 1 e 3 representam bem o cumprimento desse requisito, visto que foram criados subrotinas específicas para criação dos objetos mesa e cadeira.

Validação do código implementado

As Figuras 1 e 2 ilustram a cena 3D obtida por meio da implementação do código descrito anteriormente.

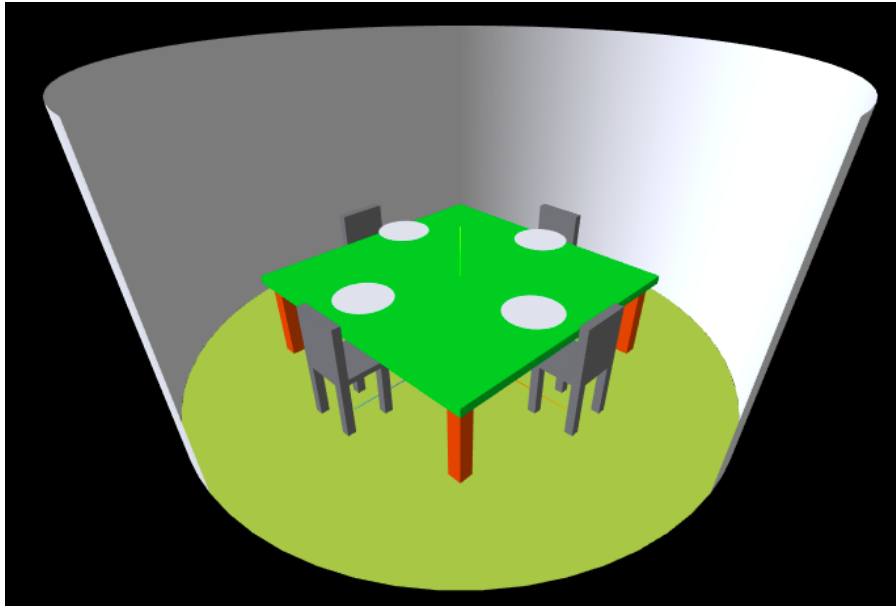


Figura 1: Cena resultante da execução do código implementado.

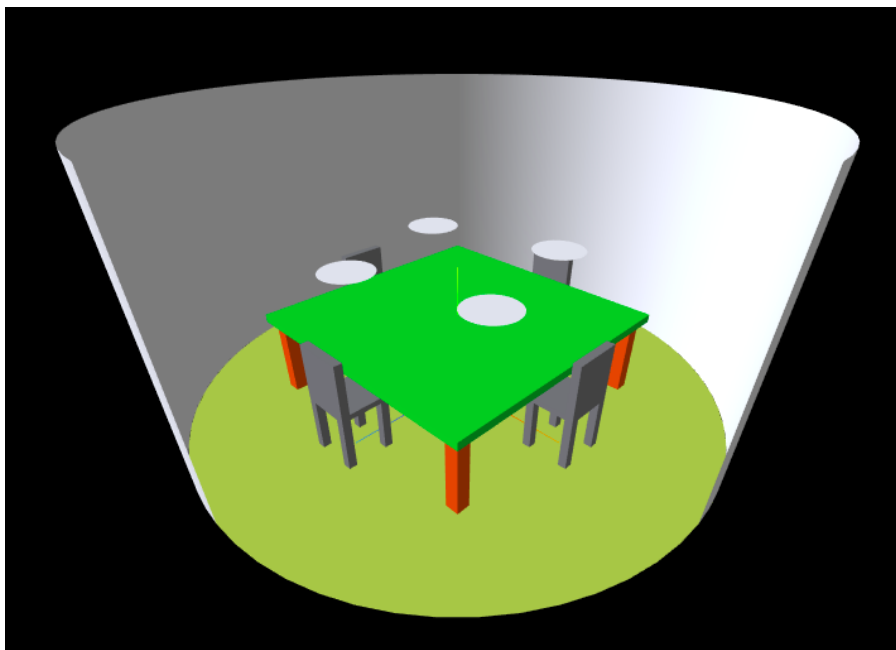


Figura 2: Figura representativa da movimentação da cena.