

Introdução ao CAD paramétrico com OpenScad

Aula 1

Por:
Eng. Prof. Dr. Carlos Eduardo Foltran

Nesta aula veremos:

- O que é CAD
- O que é desenho paramétrico
- O que é OpenScad
- Introdução ao OpenScad
- Formas primitivas
- Transformações e operações
- `cube()`, `union()` e `translate()`
- Variáveis
- Parametrização

O que é CAD?

CAD é a sigla para Computer Aided Design

Significa: desenho auxiliado por computador

OpenScad, AutoCad, TinkerCad, Blender, Fusion360, Paint, etc, são programas de CAD.

Ferramentas de CAD estão presentes mesmo em editores de texto.

O que é desenho paramétrico?

É uma técnica de CAD que permite alterar as dimensões de um desenho **sem edição do documento original**.

- Diminui o tempo necessário para alterações
- Exige software apropriado
- Demanda conhecimento especializado

O que é OpenScad?

- É um software maduro e de código aberto (free) para modelagem 3D.
- A maioria dos softwares CAD paramétricos são proprietários.
- Outras opções gratuitas, como FreeCad, são instáveis.

Vantagens do OpenScad

- Comunidade grande e ativa.
- Vasta quantidade de bibliotecas prontas.
- Facilidade de reutilização de blocos prontos (bibliotecas).
- Gera com facilidade formas matemáticas.
- É suportado pelo Thingiverse com customizadores online.
- Software de escolha para quem tem familiaridade com programação.

Desvantagens do OpenScad

- Abordagem única no gênero.
- Gera apenas o modelo 3D, sem exportação de cortes bidimensionais.
- Não é intuitivo. O aprendizado requer estudo formal.
- Não é adequado para desenhar “personagens”

Formas primitivas

O desenho é feito comendo-se formas geométricas simples (primitivas)

2D

- square()
- circle()
- polygon()

3D

- cube()
- cylinder()
- sphere()
- polyhedron()

Transformações

Transformações apenas mudam a posição ou proporção da forma geométrica no espaço.

- `translate()`
- `rotate()`
- `scale()`

Operações

Operações agem sobre duas ou mais formas geométricas para criar uma nova.

- `union()`
- `difference()`
- `intersection()`
- Etc

Primitiva cube()

- A forma primitiva cube() cria um paralelepípedo.
- O parâmetro para criação da forma é um vetor com as dimensões desejadas.
 - `cube([10,40,90]);`
- Usando o parâmetro 'center', o paralelepípedo é desenhado com centro na origem.
 - `cube([10,40,90],center=true);`

Operador union()

- Une duas ou mais formas geométricas.
 - `union(){`
 `cube([20,20,10]);`
 `cube([10,10,20]);`
 `}`
- Após uma transformação ou repetição, o operador `union()` é subentendido.

Transformação translate()

- A transformação translate() translada uma figura geométrica nas direções X, Y e Z segundo o vetor passado como parâmetro.
 - ```
union(){
 cube([20,20,10]);
 translate([5,5,0]) cube([10,10,20]);
}
```

# translate() union() {}

- Para transladar um conjunto de formas geométricas, utiliza-se o operador union().
  - ```
translate([10,10,10]) union(){  
    cube([20,20,10]);  
    cube([10,10,20]);  
}
```
- Como o operador union() é subentendido, o seguinte código produz o mesmo resultado:
 - ```
translate([10,10,10]){
 cube([20,20,10]);
 cube([10,10,20]);
}
```

# Translações concatenadas

- Pode-se concatenar translações:
  - ```
translate([10,10,10]){  
    cube([20,20,10]);  
    translate([5,5,0]) cube([10,10,20]);  
}
```

Exercício 1:

Criar uma caixa com dimensões externas de 20x20x20mm e espessura de 2mm usando `cube()`, `union()` e `translate()`.

Solução do exercício 1

//base

cube([20,20,2]);

//lateral esquerda

cube([2,20,20]);

//lateral direita

translate([18,0,0]) cube([2,20,20]);

//frente

cube([20,2,20]);

//fundo

translate([0,18,0]) cube([20,2,20]);

Variáveis

- Variáveis são nomes aos quais valores são atribuídos.
 - `A=3;`
 - `Largura=5.8;`
 - `Usuario="João"`
- As variáveis podem ter seus valores atribuídos diretamente ou calculados a partir de variáveis pré existentes. No exemplo, `Area` assumirá o valor 12.
 - `Lado1 = 3;`
`Lado2 = 4;`
`Area = Lado1 * Lado2;`
- Para confirmar o valor de uma variável, usa-se o comando `echo`:
 - `A=3;`
`echo(A)`

Variáveis

- Variáveis podem conter dados numéricos, sequência de caracteres ou listas de valores.
 - Valor = 55.9;
 - Palavras = "Isto é uma sequência de caracteres";
 - Lista = [6, 5.8, "Palavras", Palavras];
- Na lista do exemplo, o penúltimo valor é a sequência de caracteres "Palavras" e o último é o valor da variável Palavras, ou seja:
 - echo(Lista);
// imprime [6, 5.8, "Palavras", "Isto é uma sequência de caracteres"]

Nota: um vetor é uma lista de 3 valores numéricos.

Variáveis

- Uma vez atribuído um valor a uma variável, este valor não pode ser alterado.
- A alteração do valor faz o OpenScad emitir um alerta.
- O último valor atribuído tem precedência:
 - `a=3;`
`echo("primeiro valor de a=", a); // imprime 4`
`a=4;`
`echo("segundo valor de a=", a); // imprime 4`

Parametrização

- Embora seja possível, colocar dimensões fixas nas formas primitivas torna trabalhosa a alteração de um desenho.
- Se usarmos variáveis para representar as dimensões de cada uma das formas primitivas que compõem um desenho, a alteração pode ser feita de forma automática.
 - `Lado = 10;`
`cube([Lado,4*Lado, 9*Lado]);`

Exercício 2:

Criar uma caixa com dimensões parametrizadas usando `cube()`, `union()` e `translate()`.

Dica: Antes de iniciar um desenho paramétrico, identifique quais são os parâmetros que definem o seu modelo.

Solução do exercício 2

```
Espessura = 2;  
Altura = 20;    //dimensão na direção Z  
Largura = 20;   //dimensão na direção X  
Comprimento = 20; //dimensão na direção Y
```

```
//base  
cube([Largura,Comprimento,Espessura]);
```

```
//lateral esquerda  
cube([Espessura,Comprimento,Altura]);
```

```
//lateral direita  
translate([Largura - Espessura,0,0]) cube([Espessura,Comprimento,Altura]);
```

```
//frente  
cube([Largura,Espessura,Altura]);
```

```
//fundo  
translate([0,Comprimento - Espessura,0]) cube([Largura,Espessura,Altura]);
```

Desafio:

Crie uma tampa para a caixa do exercício 2 que encaixe por dentro e já considere uma folga.

Dica: Considere a folga como um parâmetro.