MC937A/MO603A – Computação Gráfica - 2021-S2 - Jorge Stolfi
Trabalho de laboratório 01 - 2021-08-18
Objeto Voador Não Imaginado

Objetivos: Treinar o uso de formas básicas e operações booleanas.

Enunciado. Depois de 70 anos de silêncio sobre o assunto de OVNIs (UFOs) o Pentágono finalmente decidiu oficialmente dizer que prefere não dizer nada a respeito. Em vista desse grande passo transversal para esse ramo da ciência, o projeto de hoje é modelar um OVNI na linguagem POV-Ray, segundo as melhores informações disponíveis na literatura.

Essa tarefa é mais fácil que que pode parecer, pois não há absolutamente nenhuma característica em comum nos retatos de que dispomos. Nem mesmo podemos dizer que eles são objetos, que voam, ou que não são identificados. Então, qualquer coisa vale...
... exceto que, para que o exercício tenha valor didático, exigimos que o seu OVNI tenha pelo menos sete elementos geométricos básicos do POV-Ray dentre `sphere`, `cylinder`, `cone` e `box`, incluindo pelo menos um de cada tipo; que seja conexo (uma peça só); e que use as operações buleanas de união e diferença, de maneira não trivial.

Parte 1. Desenhe, à mão livre, um esboço do seu OVNI, e mostre o desenho para o professor.

Parte 2. Apenas depois que o professor aprovar seu esboço: Copie os arquivos da aula passada para uma nova sub-pasta `2021-08-18` da pasta `mc937` no seu computador. Edite o arquivo `main.pov`, trocando a cena da aula anterior pela descrição de seu OVNI. Use o comando `make` para produzir uma imagem do mesmo.

Exportação. Não se esqueça de exportar seu arquivo `main.pov` até o final da aula para sua pasta WWW pública

`http://students.ic.unicamp.br/~ra`SEU_RA`/mc937-2021-2/2021-08-18/`

Originalidade. O arquivo de descrição `main.pov` deve ser construído manualmente, com um editor de texto comum, **sem** o auxílio de qualquer editor gráfico ou outra ferramenta de modelagem geométrica. Não é permitido copiar ou incluir quaisquer arquivos POV-Ray além dos fornecidos pelo professor ou escritos por você mesmo. Porém, é permitido re-usar arquivos ou trechos de código de exercícios anteriores.

Individualidade. Lembre-se de que todo trabalho prático é **individual**. Não é permitido pedir qualquer tipo de ajuda a colegas ou outras pessoas. Dúvidas devem ser tiradas apenas com o professor.

(continua)

Comandos. Os comandos POV-Ray que produzem os sólidos necessários são:
Comandos. Os comandos POV-Ray que produzem os sólidos necessários são:

- `sphere{ <` Xc`,` Yc`,` Zc`>,` R `texture{` Tx `} }`
  Este comando acrescenta à cena uma esfera. O centro é o ponto de coordenadas cartesianas (Xc, Yc, Zc), e o raio é R.

- `cylinder{ <` Xa`,` Ya`,` Za`>, <` Xb`,` Yb`,` Zb`>,` R `texture{` Tx `} }`
  Este comando acrescenta um cilindro em posição arbitrária. O ponto de coordenadas cartesianas (Xa, Ya, Za) é o centro de uma das bases, (Xb, Yb, Zb) é o centro da outra base, e R será o raio do cilindro. Os dois centros definem o eixo do cilindro.

- `cone{ <` Xa`,` Ya`,` Za`>,` Ra`, <` Xb`,` Yb`,` Zb`>,` Rb `texture{` Tx `} }`
  Este comando acrescenta um cone truncado, em posição arbitrária. O ponto (Xa, Ya, Za) é o centro de uma das bases do cone, Ra é o raio dessa base, (Xb, Yb, Zb) é o centro da outra base, e Rb é o raio dessa outra base. Use $Ra > 0$ e $Rb = 0$ para obter um cone inteiro com vértice em (Xb, Yb, Zb).

- `box{ <` Xa`,` Ya`,` Za`>, <` Xb`,` Yb`,` Zb`>` `texture{` Tx `} }`
  Este comando acrescenta uma caixa com lados paralelos aos eixos X, Y e Z. O ponto (Xa, Ya, Za) é um canto qualquer da caixa, e (Xb, Yb, Zb) é o canto oposto. Ou seja, a caixa vai de Xa até Xb na direção X, de Ya até Yb na direção Y, etc..

Em todos os comandos acima, Tx deve ser o nome de uma "tinta" definida previamente com `#declare` Tx `= texture{ ... }`. Veja o arquivo `main.pov` da aula anterio. Note que não há vírgula antes de `texture` Note também que, em POV-Ray, as coordenadas de pontos se escrevem como `<10,20,30>` e não como `(10,20,30)`.
Em cada um dos comandos acima, o modificador `translate{ <` Xt`,` Yt`,` Zt `>` } pode ser inserido antes do `texture`.