



COMPUTAÇÃO GRÁFICA



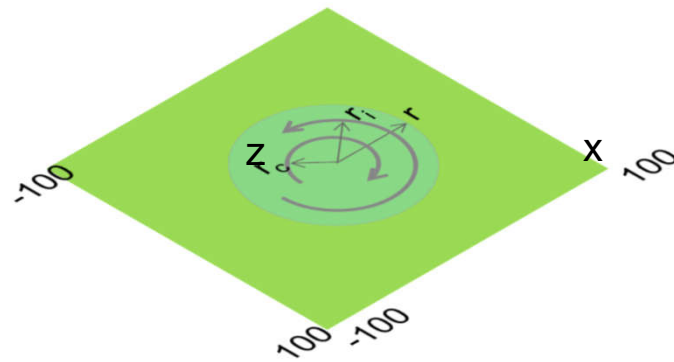
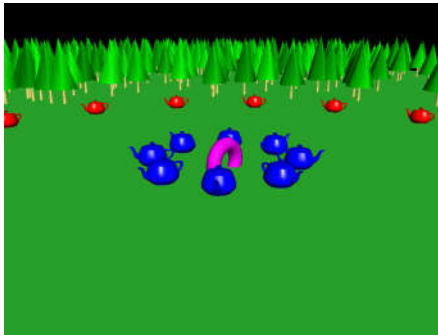
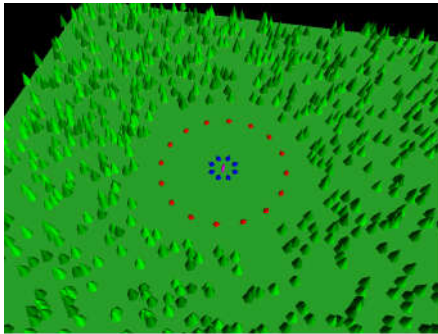
MIEI / LCC
DEPARTAMENTO DE INFORMÁTICA
UNIVERSIDADE DO MINHO

Geometric Transformations

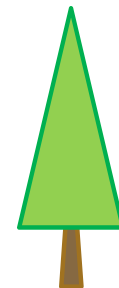
Building a 3D World



West World



$$\begin{aligned} r &= 50 \\ r_i &= 35 \\ r_c &= 15 \end{aligned}$$





Descrição mais detalhada

- O cenário está assente num plano com $Y = 0$, e com as coordenadas X e Z a variar entre -100 e 100 .
- As árvores estão dispostas no plano, excepto no círculo centrado na origem e raio 50 .
- Para desenhar uma árvore (dois cones, um em cima do outro) é necessário determinar dois números aleatórios e verificar se caem dentro do plano, mas fora do círculo central. A técnica para gerar as posições das árvores pode ser a técnica da rejeição, ou seja geram-se dois números entre -100 e 100 , e aceita-se a posição se a árvore estiver fora do círculo. Atenção que os números gerados têm de ser reais, não inteiros. Caso contrário as arvores ficarão dispostas como se tratasse de uma grelha.
- No centro do plano está um torus, tal como ilustrado na imagem (as funções para desenhar as várias primitivas encontram-se no último slide).
- Estejam atentos à disposição dos teapots e ao seu movimento. Vejam a aplicação que está no blackboard.
- O objectivo é replicar o comportamento desta aplicação.



Required functions

```
(stdlib.h)
void srand(int seed); // start the random number sequence
int rand(void); // returns a number between 0 and RAND_MAX (largest integer returned by rand())

void glTranslatef(float x, float y, float z);
void glRotatef(float ang, float x, float y, float z); // ang in degrees

void glPushMatrix(void);
void glPopMatrix(void);

void glutSolidCone(float baseRadius, float height, int slices, int stacks);
void glutSolidTorus(float innerRadius, float outerRadius, int sides, int rings);
void glutSolidTeapot(float dimensao);

glBegin(GL_TRIANGLES);
    glVertex3f(...);
    ...
glEnd();
```

