

Licenciatura em Engenharia Informática

CG - P6 Transformações

S.M. Jesus
(sjesus@ualg.pt)

FCT - Universidade do Algarve, Campus de Gambelas,
8005-139 Faro, Portugal

4 a 8 de março de 2024

[†]adaptado de *Developing Graphics Frameworks with Python and OpenGL*, Stemkoski & Pascale, CRC Press, 2022

P6 - Transformações

Objetivos

- transformações de objetos com matrizes
- interação com o utilizador
- analisar o código fornecido
- exercícios propostos

Teste e código:

- descarregar P6src.zip e correr os exemplos em transform.py, obtendo os resultados ao lado. Distinguir transformadas globais e locais.
- análise do código: ciclo de update, *Class Matrix* identificando todos os parâmetros de input.

Exercícios propostos:

- *p6-1.py*: a tecla "b" reflete o objeto em relação ao eixo x e "c" reflete em relação a y (transformação global)
- *p6-2.py*: usando uma ferramenta de modelação (recomendamos o *Blender*) realize o modelo de um **cuboide** e importe-o substituindo o triângulo deste trabalho (ver instruções abaixo). Teste as funcionalidades.
- *p6-3.py*: de forma idêntica realize o modelo do "seu" **objecto de praia da lista enviada por mensagem para o seu grupo**; importe os dados, substitua o cuboide do exercício anterior e teste as funcionalidades.

⇒ entregar na tutoria, ficheiro **P6out.zip**

com:

- (1) pasta *core* com respetivos ficheiros;
- (2) sources a executar dos exercicios propostos *p6-*.py*;
- (3) data limite de entrega **5 min antes do final da aula**.

Leitura de modelos do blender: exemplo cuboid

Parâmetros de exportação no blender

- 1 selecione o modelo a exportar (cuboid) e o formato *wavefront obj*;
- 2 recorde de ajustar os eixos na aba *transform* e selecione *Triangulate Faces*;
- 3 faça o *export* para a pasta corrente;

Leitura dos dados do modelo

- 1 coloque o ficheiro `obj_reader.py` na pasta `core`;
- 2 teste manualmente correndo `python core/obj_reader`, inserindo o nome do ficheiro `obj` apropriado e verifique que os vértices são lidos corretamente;
- 3 altere o seu ficheiro `transforms.py` com a referência do `obj_reader`

```
from core.obj_reader import my_obj_reader
```

e substitua os dados de posição dos vértices por

```
### Set up vertex attribute: reading the blender model  
position_data = my_obj_reader('[nome].obj')
```

- 4 se o objecto encher a janela de visualização, tecle "x" para afastar;