

Introduction to Python and pyPLaSM

Computational Visual Design Laboratory
(<https://github.com/cvlab>) “Roma Tre” University, Italy

Computational Graphics – Lecture 9 – March 18, 2013

Examples

Examples

Use of MKPOL constructor

MKPOL: stands for **MaKe POLyhedron**

the definition of a single convex cell (with 5 vertices)

```
verts = [[0,0],[4,0],[4,4],[2,6],[0,4]]
cells = [[1,2,3,4,5]]
pols = None
muro = MKPOL([verts, cells, pols])
VIEW(muro)
VIEW(SKELETON(1)(muro))
```

Use of primitive (translation) tensor

Translation: `T(coords)(parameters)(object)`

two primitive objects

```
door = CUBOID([1,3])  
window = CUBOID([1,1.5])
```

one assembly

```
VIEW(STRUCT([muro, door, window]))  
VIEW(SKELETON(1)(STRUCT([muro, door, window])))  
VIEW(SKELETON(1)(STRUCT([muro, T(1)(1.5)(door),  
    T([1,2])([2.75,1.5])(window)]))))
```

Use of STRUCT primitives

STRUCTture: used to make an assembly of geometrical objects

introduced the COLOR(color)(object) primitive

```
house = STRUCT([muro, COLOR(RED)(T(1)(1.5)(door)),  
               COLOR(GREEN)(T([1,2])([2.75,1.5])(window))])  
VIEW(house)
```

Use of PROD primitive

PROD: used to make the Cartesian product of geometrical objects (pointsets)

Cartesian product times an interval of size 4

```
house3D = PROD([house, Q(4)])    # properties (color) are lost  
VIEW(house3D)
```

new assembly

```
muro = PROD([muro, Q(4)])  
door = T(1)(1.5)(PROD([door, Q(4)]))  
window = T([1,2])([2.75,1.5])(PROD([window, Q(4)]))  
house = STRUCT([muro, COLOR(RED)(door), COLOR(GREEN)(window)])  
VIEW(house)
```

Advanced use of affine tensors within an assembly

PROD: used to make the Cartesian product of geometrical objects (pointsets)

$$\text{STRUCT}([Q, hpc_1, Q, hpc_2, \dots, Q, hpc_n) \equiv$$
$$\text{STRUCT}([Q(hpc_1), Q^2(hpc_2), \dots, Q^n(hpc_n))$$

```
pair_x = [T(1)(4), house]
houseRow = STRUCT(NN(10)(pair_x))
VIEW(houseRow)
```

assembly of assemblies

```
pair_z = [T(3)(14), houseRow]
neighbourhood = STRUCT(NN(10)(pair_y))
VIEW(neighbourhood)
```