

EI3-Semestre 1/IHM-RV/TD n°3

Objectif: Transformations géométriques

Éléments fournis :

- Vidéo du résultat attendu
- Code source du programme de départ (initialisation, affichage repère 3D, variables globales utiles pour l'exercice)

L'objectif est de compléter ce programme en ajoutant et/ou adaptant des fonctions utilisant et/ou modifiant les variables globales proposées. On rappelle la structure élémentaire d'une transformation simple (voir exemple de listing ci-après) :

```
glPushMatrix();  
//Transformation utilisant (glRotatef(...) ou glTranslatef(...) ou glScalef(...))  
//Primitive(s) géométrique(s)  
glPopMatrix();
```

```
from OpenGL.GL import *  
from OpenGL.GLUT import *  
from OpenGL.GLU import *  
  
def DisplayFunc():  
    glClear(GL_COLOR_BUFFER_BIT)  
    glMatrixMode(GL_MODELVIEW)  
    glPushMatrix()  
    glRotate(25,0,0,1)  
    glutWireCube(1) #On dessine un cube  
    glPopMatrix()  
    glutSwapBuffers()  
  
#MAIN  
glutInit(sys.argv)  
glutCreateWindow("")  
glutDisplayFunc(DisplayFunc)  
glutMainLoop()
```

Exemple de listing basique d'affichage d'un cube ayant subi une rotation de 25° autour de l'axe (0,z)

Pour ce travail, on utilisera la documentation en ligne. Pour chacun des appels à une transformation élémentaire (`glRotatef()`, `glTranslatef()` et `glScalef()`), on utilisera les variables globales **sans en modifier leur valeur (i.e. les valeurs initiales sont conservées tout au long de TD)**. On suivra les étapes suivantes:

- 1) Transformations élémentaires T1, T2 et T3: compléter le menu et la fonction `DrawObjects()` pour ces transformations. Il s'agit de 3 transformations élémentaires (une transformation élémentaire étant une rotation, une translation ou un changement de taille).
- 2) Transformations composées simples T4 et T5: compléter le menu et la fonction `DrawObjects()` pour ces transformations. Dans chaque cas, de quelle composition s'agit-il ? Donnez la formule utilisant l'opérateur de composition (opérateur « o », e.g. `f1 o f2`) et la formulation matricielle (i.e. ordre du produit des matrices, sans détailler leur contenu).
- 3) Transformation composée complexe T6: compléter le menu et la fonction `DrawObjects()` pour ces transformations. Indication: on utilisera une boucle pour répéter les opérations de transformation et d'affichage de l'objet élémentaire.
- 4) Transformation composée complexe T7: compléter le menu et la fonction `DrawObjects()` pour ces transformations. Indication: on utilisera une boucle pour répéter les opérations de transformation, d'affichage de l'objet élémentaire et deux empilements/dépilements de matrice (i.e. `push/popMatrix()`). Raisonner en considérant que l'on affiche d'abord une ligne de cube et que l'on répète cet affichage (i.e. plusieurs lignes de cube) en incluant la transformation adéquat. On utilisera la variable globale `nbLines`.
- 5) Bonus: utilisez les touches +/- pour augmenter/diminuer le nombre de lignes dans la question précédente, ceci ne fonctionnant que lorsque le mode (MODE) est T7.