

Enseignant(s)

**BIANCHINI Marc**

Email(s)

[mbianchi@myges.fr](mailto:mbianchi@myges.fr)

## Robot animé se déplaçant dans une scène graphique

### 1 Matières, formations et groupes

Matière liée au projet : **Introduction à l'infographie**Formations : **2I - Classe 2 || Classe 1 ||**Nombre d'étudiant  
par groupe :**2 à 3**Règles de constitution des groupes: **Libre**Charge de travail  
estimée par étudiant : **30,00 h**

### 2 Sujet(s) du projet

Type de sujet : **Imposé**

## Robot animé se déplaçant dans une scène graphique

Le sujet se décompose en trois parties:

- 1) Création d'un robot animé
- 2) Navigation de la caméra: souris/clavier
- 3) Création d'une scène graphique texturée, éclairée et munie d'autres objets apportant du réalisme

Voici les orientations à suivre:

- 1) utilisation des routines d'OpenGL afin de créer une scène graphique munie de sa caméra. Faire une ébauche de déplacement de caméra.
- 2) Création d'un robot, de votre choix, en fil de fer ou plein, dans un premier temps sans texturage. Faites une fonction pour chaque éléments répétitifs de votre robot.  
En utilisant la pile de matrices, recoller, au niveau des articulations sur le buste, les différents éléments générés par les fonctions associées.
- 3) Animer le robot: mouvement des jambes, roues, tête, bras télescopique, etc..(selon votre choix). Puis faites-le se déplacer. Éventuellement, ajouter d'autres éléments dans la scène graphique.
- 4) Utiliser des matériaux différents en fonction des éléments créés. Ajouter des sources lumineuses ponctuelles, directionnelles suivant les besoins. Pour l'illumination des parallélépipèdes, utiliser les normales en chaque vertex.
- 5) Il faudra faire une recherche personnelle sur la navigation spatiale de la caméra dans l'espace 3D à la souris et au clavier:  
Vous pouvez faire 2 types de caméra:
  - a) Libre, avec déplacement du point de vue.
  - b) Centrée, sans déplacement du point de vueDans les deux cas, utiliser les coordonnées sphériques afin de modifier les paramètres de gluLookAt().

Rq: cette partie est plus axée sur les maths. Prévoir du tps.

01/04: Livrable -> deux notes.

- 6) Prise en main du prgm SOIL qui sera fourni le 01/04. L'adapter à votre code.
- 7) Texturer le robot et les différents objets constituant la scène graphique
- 8) Effectuer une animation Robot/scène.
- 8) La scène graphique que vous allez créer est totalement libre, essayer d'être original!

### 3 Détails du projet

#### Objectif du projet (à la fin du projet les étudiants sauront réaliser un...)

Ce projet a pour but de développer l'esprit scientifique de l'étudiant suivant 3 axes majeurs:

- 1) La mise en application d'outils mathématiques, traités en cours, nécessaires à la création d'un univers 3D illuminé et texturé sous OpenGL
- 2) L'amélioration de l'apprentissage du développement en C/C++
- 3) S'adapter aux besoins du projet en effectuant des recherches personnelles en modélisation mathématiques et en utilisant, au mieux, les ressources d'OpenGL

#### Descriptif détaillé

#### Ouvrages de référence (livres, articles, revues, sites web...)

OpenGL 2.0  
Guide officiel pour l'apprentissage et la maîtrise d'OpenGL 2.0  
Mason Woo, Jackie Neider, Tom Davis et Dave Shreiner  
CampusPress

Algorithmes pour la synthèse d'images et l'animation 3D  
Rémy Malgouyres  
Dunod

#### Outils informatiques à installer

Environnement: Code::blocks  
Compilateur: MinGW  
OpenGL: librairie Glut

### 4 Livrables et étapes de suivi

1	Etape intermédiaire	Constitution des groupes, par 2 ou 3 maximum	jeudi 12/03/2015 23h30
2	Etape intermédiaire	Effectuer les parties 1),2),3),4),5) du projet. 2 notes seront attribuer: 1) Robot, animation, ébauche scène graphique 2) Matériau, Lumières, normales, caméra  Validation en classe entière.	mercredi 01/04/2015 20h00
3	Rendu final	Soutenance: 1) pwt 2) Démo 3) Quest/Rep	

---

**5****Soutenance**Durée de présentation  
par groupe :**20 min**Audience : **A huis clos**

Type de présentation :

**Présentation / PowerPoint - Démonstration**

Précisions :

**Soutenance: 1) pwt 2) Démo 3) Quest/Rep**