

TD OpenGL

L'objectif de ce TD est de compléter la scène réalisée dans le TD précédent en ajoutant des matériaux et des textures pour donner une apparence plus réaliste à la scène.

Matériaux :

- créez et paramétrez une source d'éclairage globale omnidirectionnelle :
 - composante diffuse blanche : `light_diffuse`
 - position : `light_position`
- Attribuer le matériau avec les composantes ambiante, diffuse, spéculaire et brillance `XX_YELLOW` au sol
- Créer un mure avec la fonction « `void plan()` » (facteur d'échelle `x,y,z` : 0.4, 0.4, 0.4) et lacer le au fon de la scène (voir image sur énoncé).
 - Appliquez-lui les mêmes matériaux attribués au sol
- Appliquer le matériau avec les composantes ambiante, diffuse, spéculaire et brillance `XX_LIGHT_BLUE` à la pince, aux articulations et à la base.
- Compléter la fonction générique `SetMaterial(~)` pour appliquer différentes composantes d'un matériau à un objet
 - Utiliser cette fonction en remplacement des attributions de matériaux réalisées précédemment.

Textures :

- Dans un premier temps étudier et commenter les fonctions :
 - `ImageLoad(~)`
 - `LoadGLTextures(~)`
- Appliquer la texture « `robot_tronc.bmp` » sur la face avant et arrière du tronc du robot. La texture doit couvrir entièrement les surfaces
- Appliquer la texture « `robot_segment.bmp` » aux segments du robot.
 - Vous devez appliquer une partie de l'image sur les segments.
 - Les coordonnées textures correspondent aux coordonnées des sommets (x10) du profil « `segment_2d` ». Il faudra toutefois mettre à l'échelle de l'image ces coordonnées.
 - Taille de l'image 255x255.
- Créer un tableau :
 - image centrale :
 - polygone `x z` : 4 x 6
 - texture : « `robo_copine.bmp` »
 - voir image pour les coordonnées textures
 - cadre
 - 4 polygones `x z` : 6 x 1
 - texture : « `robo_copine.bmp` »
 - voir image pour les coordonnées textures
 - Placer le cadre sur le plan arrière (voir image plus bas)
- Appliquer la texture « `robot_tête.bmp` »
 - Coordonnées de textures automatiques
- Créer une suite de sphère à placer les sur les coté de la scène (voir image plus bas)
 - Utiliser la fonction : `glutSolidSphere(Rayon,N_H,N_V)`
 - Appliquer-lui la texture « `texture_environnement.bmp` »
 - Les coordonnée texture doivent être générées automatiquement avec la fonction
 - `glTexGeni(~)`
 - paramètre d'application de la texture : `GL_SPHERE_MAP`



