

## Infographie, exercice série 7

Construire un cône à l'aide des primitives `GL_TRIANGLE_FAN`.

On utilisera 16 triangles pour dessiner la base du cône (alternativement vert et rouge), et 7 triangles pour les génératrices. Le 8<sup>ème</sup> triangle est supprimé afin de pouvoir voir l'intérieur du cône. Le sommet du cône doit se trouver à +2 unités sur l'axe des Z, la base est un cercle centré à l'origine, de rayon 1 unité, dans le plan XY. Le cône doit pouvoir tourner autour de l'axe des X avec le bouton gauche de la souris, et autour de l'axe des Y avec le bouton droit.

On utilisera deux Call Lists pour dessiner le cône et les axes.

Si on appuie sur la touche « Espace », le programme effectue une rotation continue autour de l'axe des X. On utilisera un timer pour cela.

La fonction *ModeDeDessin* ci-dessous sera utile pour représenter le cône de différentes manières.

```
void ModeDeDessin(int mask)
{
    // Turn culling on if flag is set
    if(mask & bCull)
        glEnable(GL_CULL_FACE);
    else
        glDisable(GL_CULL_FACE);

    // Enable depth testing if flag is set
    if(mask & bDepth)
        glEnable(GL_DEPTH_TEST);
    else
        glDisable(GL_DEPTH_TEST);

    // Draw back side as a polygon only, if flag is set
    if(mask & bOutline)
        glPolygonMode(GL_BACK, GL_LINE);
    else
        glPolygonMode(GL_BACK, GL_FILL);

    if(mask & bShadeModel)
        glShadeModel(GL_SMOOTH); // Interpolation de la couleur
                                // entre les différents sommets
    else
        glShadeModel(GL_FLAT);   // La couleur du polygone est la couleur
                                // du dernier sommet
}
```

`glEnable(GL_CULL_FACE);` permet d'éviter qu'une des faces du polygone soit calculée dépendant de l'appel précédent **`glCullFace(GL_FRONT)`** ou **`glCullFace(GL_BACK)`**, le défaut étant `GL_BACK` (augmentation des performances).

`glEnable(GL_DEPTH_TEST);` permet d'activer le Z-buffering (calcul des faces cachées).

`glPolygonMode(GL_BACK, GL_LINE);`  
`glPolygonMode(GL_BACK, GL_FILL);`  
permet de dessiner la face arrière du polygone en mode filaire ou en mode rempli.

`glShadeModel(GL_SMOOTH);` interpolation de la couleur entre les différents sommets.

`glShadeModel(GL_FLAT);` La couleur du polygone est la couleur du dernier sommet.

Par défaut, la face avant d'un polygone est celle qui est dessinée en tournant dans le sens contraire des aiguilles d'une montre. On peut changer cet état par la fonction :

`glFrontFace(GL_CW);` la face avant est celle où le polygone est dessiné dans le sens des aiguilles d'une montre.

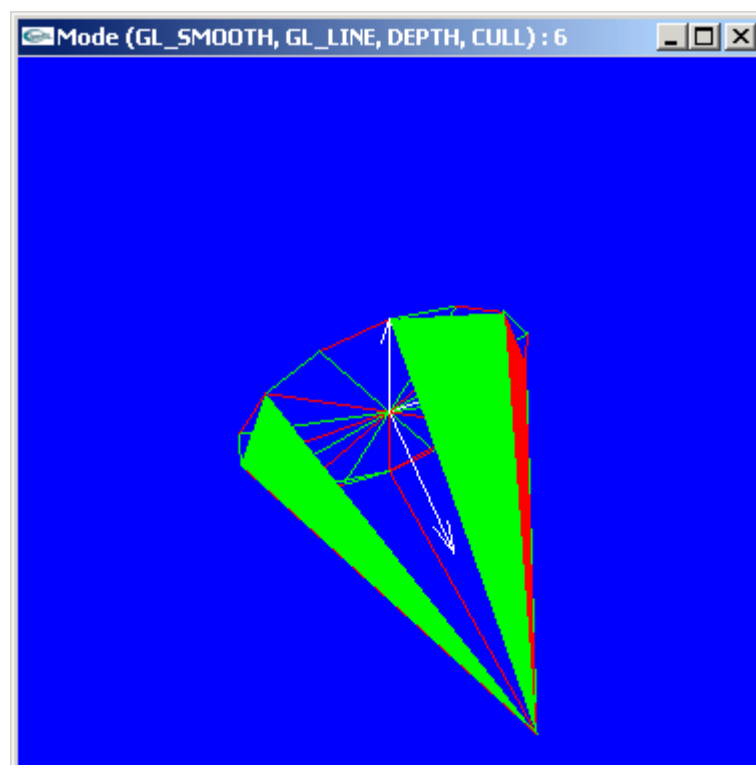
`glFrontFace(GL_CCW);` la face avant est celle où le polygone est dessiné dans le sens contraire des aiguilles d'une montre (valeur par défaut).

On peut réaliser des commandes actionnées par des touches du clavier grâce à la fonction de la librairie GLUT `glutKeyboardFunc(Keyboard)`; où la fonction **Keyboard** est une fonction de rappel dont le prototype est :

**`void Keyboard(unsigned char key, int x, int y );`**

Pour afficher la fenêtre en plein écran, on utilise : **`glutFullScreen();`**

Pour modifier la taille de la fenêtre, on utilise : **`glutReshapeWindow(500, 500);`**  
**`glutPositionWindow(30, 30) ;`**



Sur l'image ci-dessus, on voit le cône dessiner en affichant les faces arrières des triangles en mode filaire et les faces avant en mode plein. Le Depth Buffer est activé.