



PROJET INGÉNIEUR INFOGRAPHIE

# Rapport

COULON Anthony ROLLAND Cyrille 29/06/2009

# **INDEX**

Introduction	3
Fonctionnement des projets	4
première partie deux dimensions	4
deuxième partie trois dimensions	5
Code source	. 6
première partie deux dimensions	. 6
deuxième partie trois dimensions	. 7
Problèmes rencontrés	. 9

### **INTRODUCTION**

Durant le second semestre de la première année en école d'ingénieur, nous avions dans le cadre du cours d'infographie, à réaliser un projet de dessin 2D et 3D. La première partie du projet permet à un utilisateur de dessiner diverses formes en deux dimensions, la seconde partie du projet permet cette fois ci de dessiner des formes géométriques en trois dimensions comme par exemple des cubes ou bien des cylindres.

Dans ce rapport, nous verrons comment utiliser les deux petits logiciels de dessin, dans une autre partie nous verrons des points techniques du code et pour finir nous verrons les problèmes rencontrés durant l'élaboration du projet.

# I) Fonctionnement des Projets

Première partie : la deux dimensions

Un premier menu déroulant sert à sélectionner le type de dessin que nous voulons réaliser. Nous avons le choix avec :

- Dessiner au crayon, qui permet de dessiner avec la souris en pressant sur le bouton et en remuant celle-ci.
- Faire des droites avec deux algorithmes différents : naïf et Foley. Il suffit de cliquer 2 fois sur l'écran pour donner les 2 points de la droite.
- Dessiner des cercles. Là encore il suffit de cliquer 2 fois sur l'écran, une première fois pour donner le centre du cercle et la deuxième pour donner le rayon du cercles.
- Dessiner des polygones. En cliquant sur l'écran on donne les différents sommets du polygone et les sommets.
- Modifier un polygone. En cliquant sur l'écran, on récupère le sommet le plus près et on modifie ses coordonnées.
- Supprimer un point du polygone. On clique sur l'écran, on récupère le sommet le plus près et on le supprime.
- Ajouter un point au polygone. Là où on a cliqué sur l'écran, cela sera un nouveau sommet rajouter au polygone.
- Fenêtrer un polygone. Permet de fenêtrer un polygone, c'est-à-dire de faire en sorte que l'utilisateur à l'aide de la souris indique 4 droites (celles d'un rectangle) et le polygone sera restreint à ces limites.

Nous avons deux autres menus déroulant, le premier permet à l'utilisateur de choisir la couleur avec laquelle il peut dessiner et le second permet de choisir la couleur de fond de l'écran.

Trois boutons sont disponibles :

- Fin du polygone qui permet de terminer un polygone, de rajouter le dernier coté.
- Remplissage figure, qui permet de remplir une figure, de colorier soit un polygone soit un cercle
- Remplissage hachuré figure, qui remplis comme le remplissage simple mais en hachuré.

Deuxième partie : la trois dimensions

Nous avons 2 menus déroulant, trois entrées de saisies de données et un bouton. Pour chaque choix de figure dans le premier menu déroulant, le type d'information à saisir dans les entrées change.

Le bouton tracer sert, une fois qu'on a saisie les données, à tracer les figures

Quand on choisit de dessiner un cube : On indique dans la première entrée la longueur d'un des coté du cube

Le cylindre : dans l'entrée une on indique le rayon du cylindre, dans l'entrée deux sa hauteur et dans l'entrée trois son nombre de mailles.

La sphère : dans l'entrée une on indique sa hauteur et dans l'entrée deux son nombre de maille

Le cône : dans l'entrée une on indique le rayon de la face circulaire, dans l'entrée deux on indique sa hauteur et dans l'entrée trois le nombre de mailles.

La maison : dans l'entrée une on indique le coté, ensuite la hauteur du toit se calcule de la même façon à chaque fois.

La rotation : on indique dans chaque entrée les angles des rotation en x, y, z respectivement dans les entrées 1, 2, 3. On peut aussi à l'aide de la souris faire tourner les figures en cliquant sur l'écran

La translation : on indique dans chaque entrée les distances de translation en x, y, z respectivement dans les entrées 1,2 ,3. On peut aussi à l'aide de la souris faire déplacer les figures en cliquant sur l'écran.

On choisit de dessiner ou de déplacer les figures dans le premier menu déroulant. Le seconde menu déroulant sert à choisir si on veut avoir un affichage simple avec des perspectives, mais on voit toutes les arrêtes, d'avoir un affichage avec seulement le z-buffer, c'est-à-dire que les arrêtes de derrière ne sont pas apparentes, ou bien d'avoir un affichage avec le z-buffer et un effet et avec un effet de lumière.

## II) Le Code Source

#### Première partie : la deux dimensions

Trois listes contenant des points sont nécessaires pour avoir ensuite le choix de dessiner plusieurs figure à la fois

- La méthode dessineCrayon permet de dessiner au crayon avec la souris. Nous avons une boucle qui a chaque tour dessine des lignes en appelant la méthode dessineLigneFoley. Elle prend en paramètre une variable de type Graphics2D
- La méthode dessineLigneFoley prend en paramètre quatre entiers et une variable de type Graphics2D. Elle suit l'algorithme Foley pour tracer des lignes.
- La méthode dessineLigneNaif est comme la méthode dessineLigneFoley, dessine des ligne mais avec l'algorithme naïf.
- La méthode dessineCercle prend en paramètre quatre entiers, une liste de points et une variable de type Graphics2D. Cette méthode permet de dessiner des cercles.
- La méthode dessinePolygone prend en paramètre une liste de point et une variable de type Graphics2D. Cette méthode permet de dessiner des polygones. Elle trace des lignes en appelant la méthode dessineLigneFoley entre chaque point contenu dans la liste.
- La méthode remplissagePolygone prend en paramètre une liste de points et une variable de type Graphics2D. Elle permet de remplir une figure dessinée. Elle calcule les intersections avec chaque coté de la figure et dessine des points en appelant la méthode dessinePoint entre chaque intersection.

- La méthode remplissageAchureFigure prend en paramètre une liste de points et une variable de type Graphics2D. Tout comme la méthode précédente elle remplie une figure dessinée mais cette fois ci en hachuré.
- La méthode determineDistance prend en paramètre 3 points et détermine la distance d'un point avec le coté contenant les deux autres points.
- La méthode determineIndiceInsertion prend en paramètre une liste de points et un point. Elle cherche le coté le plus près du point.
- La méthode fenetragePolygone prend en paramètre deux listes de points et une variable de type Graphics2D. Elle permet de fenêtrer un polygone. Elle appelle les méthodes fenetrerUp, fenetrerDown, fenetrerLeft, fenetrerRight qui permettent de récupérer les points à conserver en haut, en bas, à gauche et à droite, après le fenêtrage. C'est à dire d'éliminer les points qui ne sont pas dans la fenêtre définie.

Deuxième partie : la trois dimensions

 La classe Maillage initialise les différentes listes qui contiendront les sommets, les mailles, les sommets projetés, les plans, les normales et les textures.

- Nous avons la méthode projeteSommets qui prend les sommets en 3 dimensions et les transforme en 2 dimensions pour être afficher par la suite.

- La méthode pinHole, de même que pour projeteSommets mais avec l'algorithme de pin hole, ce qui nous permet d'avoir afficher les perspectives.

- La méthode malle2Polygone de récupérer dans une liste les sommets projetés d'une certaine maille.

- La méthode translation permet de déplacer une maille. On appelle la méthode translationPoint3D de la classe Point3D pour cela.

- La méthode rotation qui permet de faire tourner sur lui même une figure. Pour cela on appelle les méthodes tourneOx, tourneOy et tourneOz de la classe Point3D.

- la méthode pointMoyen calcule le point moyen ou bien le barycentre de la figure.

- la méthode equationPlan calcule l'équation des mailles d'une figure.

- la méthode altitude calcule l'altitude, c'est à dire la distance entre un point et le plan de projection.

- Nous avons différentes classes qui permettent de définir des figures : Cone, Cube, Cylindre, Maison, Sphere. Ces classes héritent tous de la classe Maillage.
- Nous avons les classes Point2D, Point3D et Point4D, qui permettent de définir des point contenant soit deux coordonnées si nous sommes en deux dimensions, soit trois coordonnées si nous sommes en trois dimensions soit 4 coordonnées si nous avons une équation de plan.
- La classe dessin3D va permettre de dessiner les figure en 3 dimensions.

- La méthode dessineMaillage prend en paramètre une variable de type Graphics2D. On commence d'abord par projeter les points puis on affiche soit de manière simple, avec le z-buffer ou bien avec le z-buffer et la lumière.

- La méthode z-buffer permet d'afficher que les faces visibles c'est à dire les faces de devant. On calcule l'altitude et on ne garde que la plus petite pour avoir que les faces visibles. On conserve les altitudes dans un tableau et les mailles visibles dans un autre tableau.

- La méthode lumiere permet de calculer la lumière. On fixe le point source de lumière et les deux intensités, ambiante et ponctuelle; enfin on calcule les lumières ambiante, ponctuelle et spéculaire.

#### III) Problèmes rencontrés

Nous avons rencontrés des problèmes avec l'algorithme du z-buffer. Nous pensions au départ à un souci de calcule de l'altitude, nous avons tous vérifier pour arriver à la conclusion que l'altitude était calculé correctement. Nous avons ensuite vérifié les calculs d'équations de plan, mais tous semblaient bons. Au final nous avons vu que notre erreur venait de la définition du point observateur. Cette correction nous a permis d'avoir un algorithme du z-buffer qui fonctionne correctement et d'avoir un meilleur rendu avec la lumière.

Nous avons eu d'autres problèmes pour l'application des textures que nous avons résolus.