

Courbes de Bézier et police de caractères

1 Informations pratiques

Ce projet est à effectuer en binômes. Dès qu'un binôme sera constitué, il enverra un mail à vtorri@univ-evry.fr pour vérification.

Le code rendu sera écrit en C++ pour le développement. **Tout projet ne compilant pas se verra attribuer la note de 0.** Mieux vaut rendre un projet incomplet mais compilant, qu'un projet ne compilant pas. Si plusieurs binômes ont des codes trop similaires, leur note sera divisée par le nombre de binômes impliqués.

Le code devra être indenté de manière uniforme. La définition des méthodes et des fonctions ne devront pas dépasser un nombre de lignes raisonnable (au plus 50 lignes)

Le code devra être commenté en utilisant la syntaxe de Doxygen.

Le rapport sera écrit en L^AT_EX et inclura les choix, les problèmes techniques qui se posent et les solutions trouvées (la conception (dont un diagramme UML complet) et réalisation). Le soin apporté à la grammaire et à l'orthographe sera largement pris en compte.

Le projet sera envoyé sous forme d'archive tar.xz ayant pour nom *NOM1_NOM2_projet_bezier.tar.xz* avant le Dimanche 13 janvier 2019 à minuit à l'adresse vtorri@univ-evry.fr. Le non respect du nom de l'archive sera pénalisé par 5 points en moins. L'archive contiendra le rapport en L^AT_EX et PDF, ainsi que les fichiers C++. **Tout projet rendu en retard se verra attribuer la note de 0.** Donc ne pas attendre le dernier moment pour l'envoyer. Un mail de confirmation sera envoyé.

2 Les courbes de Bézier

Les polices de caractères TrueType ont été introduites dans les années 1980 pour palier aux limitations des polices existantes jusqu'alors : les polices bitmaps. En effet, celles-ci étaient constituées d'un bitmap monochrome (rectangle de pixels de couleur blanc ou noir). Ainsi, pour chaque taille de caractère, par exemple, il fallait un fichier bitmap. Ce qui augmente entre

autre la taille des fichiers sur disque dur d'autant.

Les polices TrueType sont elles définies avec des courbes géométriques : les courbes de Bézier (du nom de leur inventeur : Pierre Bézier). Plus précisément des courbes de Béziens linéaires et quadratiques.

Courbes de Bézier linéaire Une courbe de Bézier linéaire est définie par deux points P_0 et P_1 et est constituée du segment $[P_0, P_1]$.

Courbes de Bézier quadratique Une courbe de Bézier quadratique est définie par 3 points P_0 , P_1 et C_{12} . Les points P_1 et P_2 sont les extrémités et le point C_{12} est le point de contrôle.

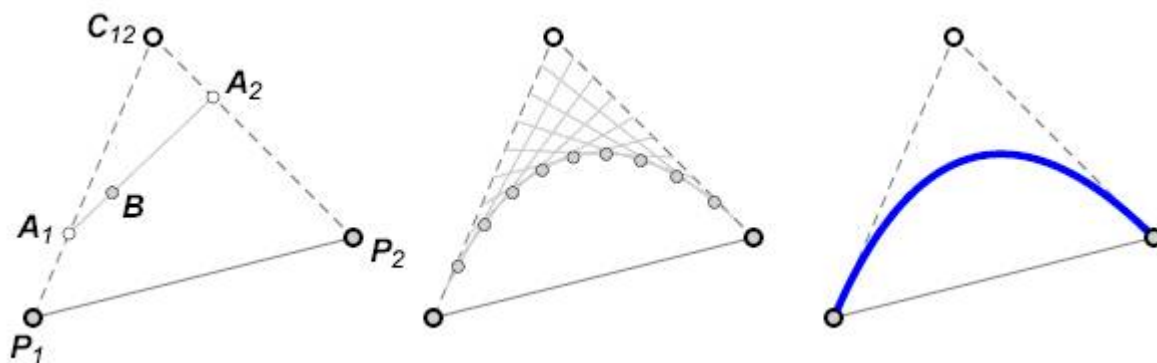


FIGURE 1 – Construction d'une courbe de Bézier quadratique

Pour construire l'ensemble des points de la courbe, on se fixe un $x \in [0, 1]$ et on construit deux points : A_1 dans le segment $[P_1, C_{12}]$ tel que la distance entre P_1 et A_1 soit égale à x fois la distance entre P_1 et C_{12} . De même pour un point A_2 dans le segment $[C_{12}, P_2]$ tel que la distance entre C_{12} et A_2 soit égale à x fois la distance entre C_{12} et P_2 . Le point $B = B(x)$ de la courbe de Bézier se situe sur le segment $[A_1, A_2]$, tel que la distance entre A_1 et B soit égale à x fois la distance entre A_1 et A_2 (voir la Figure 1).

Ainsi un cercle peut être défini par un ensemble de 8 courbes de Bézier (voir la figure 2 page 3).

Police de caractère et glyphe Une police de caractère est donc constituée d'un ensemble de caractère, chacun étant défini par son contour : le *glyphe*. Ce dernier est défini par un ensemble de courbes de Bézier. Le glyphe de la lettre "0" peut être par exemple formé de deux courbes de Bézier formant chacune une courbe proche d'une ellipse).

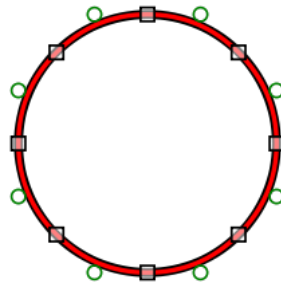


FIGURE 2 – Les carrés : extrémités. Les cercles : points de contrôle

3 Algorithme de de Casteljau

Pour tracer une courbe de Bézier, on n'utilisera pas la formulation mathématique présentée dans la section précédente, mais un algorithme récursif développé par le mathématicien Paul de Casteljau.

Le principe est de construire récursivement une ensemble de barycentres se rapprochant au fur et à mesure de la courbe. L'algorithme s'arrête par exemple quand toutes les distances entre deux points consécutifs sont suffisamment petites.

4 Sujet

Ecrire un programme en C++ correctement structuré, sans l'aide d'une quelconque bibliothèque extérieure au C++, qui définira trois polices de caractères pour les lettres de l'alphabet en majuscule et sans sérif uniquement :

1. La première police consistera à tracer uniquement les glyphes décrivant le contour du caractère en noir dans un bitmap (rectangle de pixels blancs et noirs, les pixels noirs étant donc le contour du caractère).
2. La deuxième police consistera à en plus remplir l'intérieur du caractère en noir, toujours dans un bitmap.
3. La troisième police consistera à rajouter un contour rouge de deux pixels à l'extérieur du caractère. Ce style de police est très utile pour les sous-titrages des films, pour pouvoir toujours lire le sous-titre quelque soit la scène du film.

Eventuellement, on pourra utiliser la bibliothèque libpng pour fournir des images png de chaque caractère.