

TD/TP n°2 : courbes paramétriques

Exercice 1 :

- ⇒ Faire une fonction qui trace une courbe cubique d'Hermite :
 - rajouter au programme du TP précédent, une fonction pour tracer avec OpenGL une courbe définie par un ensemble de points : `void DrawCurve(Point TabPointsOfCurve [], long nbPoints) ;`
 - signature de la nouvelle fonction : `void HermiteCubicCurve(Point P0, Point P1, Vector V0, Vector V1, long nbU, Point *resultat) ;`
 - tester votre fonction avec les points P0(0,0) et P1 (2,0) et les vecteurs V0(1,1) et V1(1,-1).

Exercice 2 :

- ⇒ Faire une fonction qui trace une courbe de Bézier par les polynômes de Bernstein :
 - signature : `void BezierCurveByBernstein(Point TabControlPoint [], long nbControlPoint, long nbU, Point *resultat) ;`
 - utiliser la fonction de l'exercice précédent pour tracer le polygone des points de contrôle et la courbe de Bézier correspondante ;
 - tester plusieurs séries de points de contrôle pour voir les différentes courbes qui en résultent. Vous pouvez utiliser les événements clavier pour modifier les points durant l'exécution du programme.

Exercice 3 :

- ⇒ Faire une fonction qui trace une courbe de Bézier par l'algorithme de Casteljau
 - signature : `void BezierCurveByCasteljau(Point TabControlPoint[], long nbControlPoint, long nbU, Point *resultat) ;`
 - tester votre fonction en affichant les étapes intermédiaires de la construction des points. **(A RENDRE au plus tard le 04/02/2019 car noté)**