

TP 2 WebGL : Courbes de Bézier en WebGL

En se basant sur l'interface en WebGL que vous avez créé, vous devez :

1. Pouvoir insérer des points de contrôle : avec les coordonnées ou avec la souris.
2. Tracer le polygone de contrôle.
3. Tracer la courbe de Bézier qui modélise ces points de contrôle.

Vous pouvez choisir de l'implémenter avec :

- la définition avec les fonctions de base de Bernstein, qui seront aussi dessinées.

ou

- l'algorithme de Decasteljau et l'affichage des pas de l'algorithme.

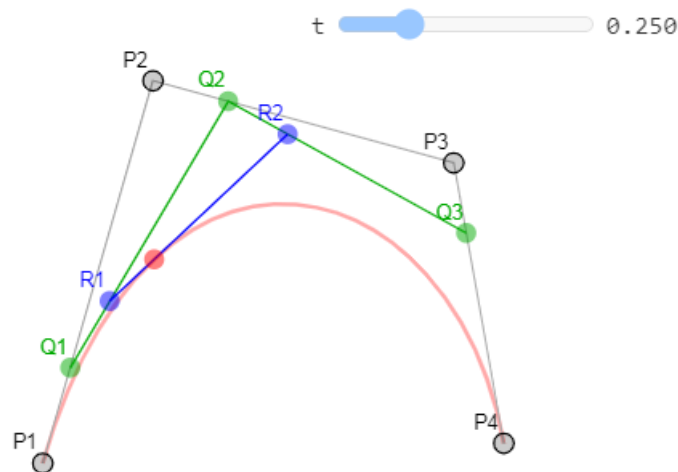
4. Pouvoir modifier les points de contrôle et afficher la nouvelle courbe de Bézier.

Applications :

- a) Représenter les trois courbes de Bézier de degré 3 dont les points de définition sont :

- $P_0 = (0, 0)$, $P_1 = (0, 1)$, $P_2 = (1, 1)$ et $P_3 = (1, 0)$
- $P_0 = (0, 0)$, $P_1 = (1, 0)$, $P_2 = (0, 1)$ et $P_3 = (1, 1)$
- $P_0 = (0, 0)$, $P_1 = (1, 1)$, $P_2 = (0, 1)$ et $P_3 = (1, 0)$

- b) (Bonus) Modifier la courbe avec des transformations géométriques (translations, homothétie, rotation, combinaisons de transformations).



À rendre :

- Code source bien commenté.
- Un manuel technique qui décrit la structure et le fonctionnement du code WebGL (par exemple si vous utilisez des libraires particulières, lesquelles et comment elles fonctionnent ...) ainsi que vos fonctions.
- Un manuel utilisateur de votre programme.