

Übung zum Praktikum Computergrafik 6

Aufgabe 1 (Bézier Kurve mit deCasteljau)

4 + 4 + 4 + 4 Punkte

Im obj file format wird eine Bézier Kurve (vereinfacht) wie folgt abgespeichert:

```
v 0.0 0.0 0.0  
v 0.0 1.0 0.0  
v 1.0 1.0 0.0  
v 1.0 0.0 0.0  
cstype bezier  
deg 3  
curv 1 2 3 4  
end
```

Mit *deg* wird der Grad der Kurve bezeichnet, $deg + 1$ ergibt die Anzahl der Bézier Kontrollpunkte. Nach *curv* stehen die Indices der Vertices, die wir als Kontrollpunkte nehmen.

Implementieren Sie die Bézier Kurve mit dem **deCasteljau Algorithmus** mit beliebiger Kontrollpunktanzahl.

Hinweis: Eine Implementierung der Bézier Kurve über Bernsteinpolynome ergibt 0 Punkte!

Gehen Sie dazu wie folgt vor:

1. Implementieren Sie das Einlesen einer Bézier Kurve aus einem (vereinfachten) obj file für eine beliebige Kontrollpunktanzahl. Stellen Sie die Kontrollpunkte durch kleine Kugeln dar und verbinden Sie benachbarte Kontrollpunkte mit Linien. **Skalieren und Transformieren Sie das eingeladene Kontrollpolygon mittig passend in ihre Szene.**
2. Visualisieren Sie die Bézier Kurve für $t \in [0, 1]$ durch Liniensegmente. Die Schrittweite, mit der Sie die Bézier Kurve absteppen, kann dabei durch die Tasten 'b' und 'B' jeweils verkleinert und vergrößert werden. Lassen Sie einen Kegel auf der Bézier Kurve laufen. Der Kegel zeigt in **Richtung der Ableitung**. Kegel und Bézier Kurve besitzen eine unterschiedliche Farbe. Die Geschwindigkeit des Kegels auf der Bézier Kurve kann mit 'k' und 'K' erhöht und reduziert werden (bis zum Anhalten des Kegels).
3. Implementieren Sie die Visualisierung der Konstruktion der Zwischenpunkte des deCasteljau Algorithmus. Die Konstruktion soll mit dem Kegel mitlaufen, die Geschwindigkeit ist ebenfalls über 'k' und 'K' zu regeln. Über Tastendruck ist die Visualisierung der Zwischenpunkte zu- und abschaltbar.

4. Implementieren Sie die vorherigen Aufgabenteile mit deCasteljau für $t \in [-1, 2]$.
Hinweis: Die Kurve verlässt die konvexe Hülle der Kontrollpunkte!

Abgabezeitraum: 24. Juni im Praktikum