

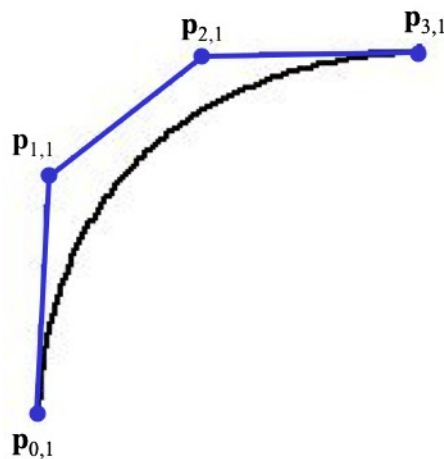
Übungsblatt 11

Willkommen zur elften Übung der Veranstaltung *Generative Computergrafik*. Ziel dieses Übungsblatts ist, dass Sie sich mit *B-Spline-Kurven* vertraut machen. Aufgabe 2 ist bis zum 10. Juli 2019 um 10:00 Uhr über <https://read.mi.hs-rm.de> abzugeben und im Praktikum vorzuführen.

Aufgabe 1. Gegeben sei die unten abgebildete kubische Bézier-Kurve

$$\mathbf{x}_1(t) = \sum_{i=0}^3 B_{i,[1,2]}^3(t) \mathbf{p}_{i,1}$$

über dem Parameterintervall $[1, 2]$. Diese sei Bestandteil einer kubischen Bézier-Splinekurve mit den Parameterintervallen $[0.5, 1]$, $[1, 2]$, und $[2, 3]$.



1. Konstruieren Sie eine Bézier-Kurve $\mathbf{x}_2(t)$ (über $[2, 3]$), welche im Punkt $\mathbf{p}_{3,1}$ zweimal stetig differenzierbar an die Kurve \mathbf{x}_1 anschließt.
2. Konstruieren Sie eine Bézier-Kurve $\mathbf{x}_0(t)$ (über $[0.5, 1]$), welche im Punkt $\mathbf{p}_{0,1}$ zweimal stetig differenzierbar an die Kurve \mathbf{x}_1 anschließt.

Aufgabe 2. Schreiben Sie ein glfw-Programm (**verwenden Sie nicht das mit Übungsblatt 10 vorgegebene Tkinter-Skript bezierTemplate.py**), welches analog zu Aufgabe 2 auf Übungsblatt 10, die Eingabe von Punkten $\mathbf{p}_0, \dots, \mathbf{p}_n$ mit Hilfe der Maus erlaubt und das Polygon durch die Punkte sowie eine Kurve zeichnet. Die dargestellte Kurve soll eine **B-Spline-Kurve**

$$\mathbf{x}(t) = \sum_{i=0}^n N_i^k(t) \mathbf{p}_i$$

der Ordnung k mit uniformen Knotenvektor (der Länge $n + k + 1$)

$$K = \{\underbrace{0, \dots, 0}_k, 1, 2, \dots, n - (k - 1), \underbrace{n - (k - 2), \dots, n - (k - 2)}_k\}$$

und (Kontroll-)Punkten $\mathbf{p}_0, \dots, \mathbf{p}_n$ sein.

Gehen Sie bei der Implementierung analog zu Teil 1 von Aufgabe 2 auf Blatt 10 vor. Schreiben Sie dazu eine Funktion

`deboor(degree, controlpoints, knotvector, t)`

die mit Hilfe des *de-Boor-Algorithmus* einen Punkt auf einer B-Spline-Kurve berechnet. Die einzelnen Parameter der Funktion haben dabei folgende Bedeutungen:

- **degree: Polynomgrad** der B-Spline-Kurve
- **controlpoints: Kontrollpunkte** der B-Spline-Kurve
- **knotvector: Knotenvektor** der B-Spline-Kurve
- **t: Parameterwert**, zu dem der Punkt auf der B-Spline-Kurve berechnet werden soll

Verwenden Sie die Funktion `deboor()` dann, um eine feste Anzahl m von Punkten auf der Kurve zu berechnen und anschließend als Polygon darzustellen. Beachten Sie, dass eine B-Spline-Kurve der Ordnung k erst dann vollständig definiert ist, wenn wenigstens k Punkte angegeben wurden. Bis zur Eingabe des k -ten Punktes soll lediglich das (Kontroll-)Polygon durch die Punkte dargestellt werden.

Ihr Programm soll es weiterhin ermöglichen die beiden Parameter k (*Ordnung* der Kurve) und m (*Anzahl* zu berechnender *Kurvenpunkte*) zu beeinflussen. Die *Ordnung* soll dabei mit den Tasten `k` bzw. `K`, die *Anzahl Kurvenpunkte* mit den Tasten `m` bzw. `M` *verringert* bzw. *erhöht* werden können.