PD. Dr. Martin Kreuzer
Fakultät für Mathematik
Universität Regensburg

WS 2001/2002

Übungen zur Vorlesung "Computeralgebra für Naturwissenschaftler"

Tutorium 5: Berechnung von Splines

Nachfolgend wollen wir einige der Resultate von Abschnitt 16 der Vorlesung in CoCoA implementieren und anwenden. Seien $a, b \in \mathbb{Q}$ mit $a < b$, sei $\mathcal{C} = (c_0, \dots, c_k) \in \mathbb{Q}^{k+1}$ eine Zerlegung des Intervalls $[a, b]$, und seien $d_0, \dots, d_k \in \mathbb{Q}$ vorgegebene Funktionswerte.

a) Für $i = 0, \dots, k$ sei $\ell_i = \prod_{j \neq i} \frac{x - c_j}{c_i - c_j} \in \mathbb{Q}[x]$. Beweisen Sie, dass das *Lagrange-Interpolationspolynom* $\ell = \sum_{i=0}^{k} d_i \ell_i$ einen Grad $\deg(\ell) \leq k$ hat und durch die Stützpunkte $(c_0, d_0), \dots, (c_k, d_k)$ geht.

b) Schreiben Sie eine CoCoA-Funktion `Lagrange(...)`, die als Eingabe eine Liste $[[c_0, d_0], \dots, [c_k, d_k]]$ erwartet und die das Lagrange-Interpolationspolynom berechnet.

c) Wenden Sie Ihre Funktion `Lagrange(...)` in den folgenden Fällen an.
   1) Interpolation von $f : [0, 2\pi] \longrightarrow \mathbb{R}$ $(t \mapsto \sin(t))$ mit Stützpunkten bei $c_i = \frac{i\pi}{2}$ für $i = 0, \dots, 4$.
   2) Interpolation von $g : [-2, 2] \longrightarrow \mathbb{R}$ $(t \mapsto \frac{1}{1+25t^2})$ mit Stützpunkten bei $(-2, -1, 0, 1, 2)$.
   3) Interpolation von f mit Stützpunkten bei $\tilde{c}_i = \frac{i\pi}{4}$ für $i = 0, \dots, 8$.
   4) Interpolation von g mit Stützpunkten bei $(-2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, 2)$.

Wie hat sich die Vergrößerung der Zahl der Stützpunkte jeweils ausgewirkt?
Hinweis: Schreiben Sie zuerst eine CoCoA-Funktion `Sin(...)`, die mit Hilfe der Taylor-Entwicklung den Wert von $\sin(t)$ bis auf eine gegebene Nachkommastellenzahl genau berechnet. Implementieren Sie dann eine Funktion `Values(...)`, die aus $[c_0, \dots, c_k]$ die Liste $[[c_0, d_0], \dots, [c_k, d_k]]$ bestimmt. Der Name der zu interpolierenden Funktion soll dabei eines der Argumente von `Values(...)` sein.

d) Schreiben Sie eine CoCoA-Funktion `Splines(...)`, die bei angegebenem $r \geq 0$ ein Erzeugendensystem des $\mathbb{R}[x]$-Moduls $\mathbb{S}^r(\mathcal{C})$ der r-mal stetig differenzierbaren $\mathcal{C}$-Splines berechnet.

e) Beweisen Sie die Formeln von Satz 16.11 für die Berechnung natürlicher Splines.

f) Implementieren Sie eine CoCoA-Funktion `NatSpline(...)`, die aus einer Liste von Stützpunkten $[[c_0, d_0], \dots, [c_k, d_k]]$ den zugehörigen natürlichen $\mathcal{C}$-Spline berechnet.

g) Wenden Sie Ihre Funktion `NatSpline(...)` an, um die Funktion $h : [0, 3] \longrightarrow \mathbb{R}$ mit $h(t) = \sin(e^t)$ und den Punkten $c_i = \frac{i}{4}$ mit $i = 0, \dots, 12$ zu interpolieren.
Hinweis: Um die Funktion $h(t)$ genügend genau zu berechnen, sollten Sie $\sin(t)$ und e^t weit genug (z.B. die ersten 70 Terme) in eine Taylorreihe entwickeln.

h) Wiederholen Sie den vorhergehenden Teil mit der Zerlegung $\mathcal{C}' = (0, 0.45, 1.15, 1.55, 1.85, 2.05, 2.25, 2.4, 2.55, 2.65, 2.75, 2.85, 3)$. Welche der beiden Spline-Funktionen approximiert h besser?