Fachbereich Mathematik
N. Borg

SS 2015
Numerische Integration

Computerpraktikum, MATLAB

Numerische Integration

Einführung 1: Sei $V \subset Abb(\mathbb{R}, \mathbb{R})$ $(n+1)$-dimensionaler linearer Unterraum und $n \in \mathbb{N}$. Weiter seien Stützstellen $\{x_i\}_{i=0}^n \subset \mathbb{R}$ sowie Zielwerte $\{y_i\}_{i=0}^n \subset \mathbb{R}$ gegeben.

Aufgabe: Finde $p \in V$ (Interpolierende) , sodass $p(x_i) = y_i, i = 1, \ldots, n$.

Einführung 2: Sei $f \in C(\mathbb{R})$. Gesucht ist eine Methode $I_h(f)$ zum numerischen Berechnen von Integralen mit beliebem Integraden f, d.h.

$$I_h(f) \approx I(f) = \int_a^b f(x)dx$$

Aufgabe 1: Arbeiten Sie sich in das Thema numerischen Integration ein.

Aufgabe 2: Implementieren Sie in MATLAB das *Siebs des Eratosthenes* für eine frei wählbare obere Schranke. Sie erhalten je nach oberer Schranke $n \in \mathbb{N}$ Primzahlen $q_i, i = 1, \ldots, n$. Ordnen Sie jeder Primzahl eine weitere Zahl $x_i, i = 1, \ldots, n$ zu (bspw. Durchnummerierung) um eine Datenmenge $(x_i, q_i), i = 1, \ldots, n$ zu erzeugen.

Aufgabe 3: Implementieren Sie in MATLAB einen Lösungsalgorithmus (Stichwort „Interpolation “) zur Bestimmung von $p \in Pol^n(\mathbb{R})$ (Raum der reellwertigen Polynome vom Grad n). Weiter soll das entstandene LGS mittels des Gauß'schen Eliminationsverfahren gelöst werden.

Aufgabe 4: Implementieren Sie in MATLAB die Newton-Codes-Formel bis zur Ordnung 6 (bei Zeitknappheit reduzieren). Hierbei wird wert auf die

Herleitung der *Gewichte* gelegt.

Aufgabe 5: Zeigen Sie anhand der in Aufgabe 3 bestimmten Polynome, dass die Newton-Codes-Formeln der Ordnung $n \in [0, \ldots, 6]$ Integrale mit Polynomen vom Grad $\leq n$ exakt berechnen.

Aufgabe 6: Implementieren Sie in MATLAB die summierten Quadratur-Formeln für beliebig viele Stützstellen und bis Ordnung 6.

Aufgabe 6: Vergleichen Sie die einfachen Quadratur-Formel aus Aufgabe 4 mit denen der Summierten aus Aufgabe 6 (Resiuumverlauf bei ansteigender Zahl von Stützstellen,...)

Bemerkung: Alle Algorithmen zur numerischen Berechung von Intergalen müssen auch für beliebige Funktionen funktionieren und nicht nur für Polynome.