MATHEMATISCHES INSTITUT
PROF. DR. CHRISTIANE HELZEL
DR. GEORG JANSING
PAWEL BUCHMÜLLER
MAXIMILIAN SCHNEIDERS
1. JUNI 2016

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

22	23	24	Σ

NAME: ____________
MAT-NR.: ____________
NAME: ____________
MAT-NR.: ____________
NAME: ____________
MAT-NR.: ____________
GRUPPE: ____________

Numerik I – 8. Übungsblatt

Aufgabe 22: (3 + 3 Punkte)

Sei

$$Q_n(f) = \sum_{i=0}^{n} a_i f(x_i)$$

eine interpolatorische Quadraturformel zum Grundintervall $[a,b]$. Zeigen Sie, dass

$$\widetilde{Q}_n(f) = \sum_{i=0}^{n} \tilde{a}_i f(\tilde{x}_i), \text{ mit } \tilde{a}_i = \frac{\tilde{b}-\tilde{a}}{b-a} a_i, \ \tilde{x}_i = \tilde{a} + (x_i - a)\frac{\tilde{b}-\tilde{a}}{b-a}$$

eine Quadraturformel auf dem Intervall $[\tilde{a}, \tilde{b}]$, ihre s.g. *Transformierte*, liefert.

Zeigen Sie dass Q_n und $\widetilde{Q}_n$ dieselbe Ordnung haben.

Aufgabe 23:

Zeigen Sie:

(a) Für alle $f \in C^2([-1,1], \mathbb{R})$ und $\alpha \in \mathbb{R}$ gilt

$$\int_{-1}^{1} f(x)\,\mathrm{d}x = f(1) + f(-1) - (\tfrac{1}{2} + \alpha)(f'(1) - f'(-1)) + \int_{-1}^{1} (\tfrac{1}{2}x^2 + \alpha) f''(x)\,\mathrm{d}x.$$

(b) Für alle kubischen Polynome $p \in \mathbb{P}_3$ gilt

$$\int_{-1}^{1} p(x)\,\mathrm{d}x = p(1) + p(-1) - \tfrac{1}{3}(p'(1) - p'(-1)).$$

(c) Für alle $a, b \in \mathbb{R}$ mit $a < b$ und alle kubischen Polynome $q \in \mathbb{P}_3$ gilt

$$\int_{a}^{b} q(x)\,\mathrm{d}x = \frac{b-a}{2}(q(b) - q(a)) - \frac{(b-a)^2}{12}(q'(b) - q'(a)).$$

Aufgabe 24:

Bestimmen Sie für das gewichtete Integral

$$I(f) = \int_{-1}^{1} f(x) \sin(\tfrac{1}{2}\pi x)\,\mathrm{d}x$$

eine Quadraturformel der Form

$$Q(f) = a_0 f(-1) + a_1 f(0) + a_2 f(1),$$

die für Polynome vom Grade kleiner oder gleich 2 exakt ist, d.h. $I(p) = Q(p)$ für alle $p \in \mathbb{P}_2$. Berechnen Sie anschließend $Q(\sin(\frac{1}{2}\pi x))$ und vergleichen das Ergebnis mit dem exakten Wert $I(\sin(\frac{1}{2}\pi x))$.

b.w.

Programmieraufgabe 8:

Für $f \in C([a,b],\mathbb{R})$ suchen wir mithilfe verschiedener Quadraturverfahren Approximationen an das Integral

$$I(f) = \int_a^b f(x)\,\mathrm{d}x. \tag{1}$$

(a) Schreiben Sie eine Funktion `v = IterTrapez(f, a, b, n)`, die die iterierte Trapezregel auf `n` Teilinterballen benutzt, um (1) zu approximieren.

(b) Schreiben Sie eine Funktion `v = IterSimpson(f, a, b, n)`, die die iterierte Simpson-Regel auf `n` Teilinterballen benutzt, um (1) zu approximieren.

(c) Schreiben Sie eine Funktion `v = Romberg(f, a, b, n)`, die das Romberg-Verfahren (d.h. Extrapolation mit der Romberg-Folge $h_0, h_1, \ldots h_n$ des Vorlesungsskriptes) benutzt, um (1) zu approximieren. Hierbei ist $\mathtt{n}+1$ die Anzahl an zu verwendenden Schrittweiten.

(d) Schreiben Sie ein Skript `p8`, dass Ihre Verfahren an $f := x \mapsto \ln(x)$ und für $\mathtt{n} = 1, 2, \ldots, 8$ und $[a,b] = [1,2]$ testet: Berechnen Sie den exakten Wert und messen Sie für alle drei Verfahren die Fehler für die angegebenen `n`. Stellen Sie diese in einem doppelt-logarithmischen Plot dar.

(e) Erweitern Sie Ihr Skript, so dass es nun die iterierte Trapezregel mit $\mathtt{n} = 2, 4, 8, 16, 32, 64, 128, 256$, die iterierte Simpson-Regel mit $\mathtt{n} = 1, 2, 4, 8, 16, 32, 64, 128$ und das Romberg-Verfahren mit $\mathtt{n} = 1, 2, 3, 4, 5, 6, 7, 8$ miteinander vergleicht. Berechnen Sie die Fehler wie im vorigen Aufgabenteil, aber plotten Sie die Daten auf der x-Achse alle gegen die `n`s der Trapezregel.

(f) Zusatz: Achten Sie darauf, dass Sie bei Ihren Programmen aus den ersten drei Teilaufgaben keine mehrfachen Funktionsauswertungen verwenden. Bauen Sie in Ihre Funktionsauswertung keine kurze Wartezeit ein (etwa 0,1 Sekunden). Plotten Sie die Fehler mit den `n`s aus dem vorigen Aufgabenteiles nun gegen mit die benötigte Zeit.

Hinweis: Sie können (`MATLAB`-/`Octave`-Syntax) die Funktion

```
function y = f(x), sleep(0.0025 * length(x)); y = log(x); end;
```

verwenden.

Der Test mit der unterschiedlichen Anzahl an Intervallen ist deutlich fairer, als der erste Test gegen eine fixe Anzahl an Schritten, da die Verfahren im Falls der unterschiedlichen `n`s (falls in der Implementierung darauf geachtet wurde) dieselbe Anzahl an Funktionsauswertungen verwendet werden.

Abgabe der Übungsaufgaben am Mittwoch, 8. Juni zu Beginn der Vorlesung.
Abgabe der Programmierübungen per E-Mail bis Freitag, 10. Juni, 23:59 Uhr an `num1@hhu.de` **mit Betreff** `PA# Gr#`**, wobei # für die Nummer der Programmieraufgabe bzw. der Programmierübungsgruppe steht.**