MATHEMATISCHES INSTITUT
PROF. DR. ACHIM SCHÄDLE
FELIX LIEDER
DR. GEORG JANSING

27. APRIL 2017

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

4	5	6	Σ

NAME: _____
MAT-NR.: _____
NAME: _____
MAT-NR.: _____
NAME: _____
MAT-NR.: _____
GRUPPE: _____

Numerik I – 2. Übungsblatt

Aufgabe 4: (4 + 2 Punkte)

Sei $p \in \mathcal{P}_2$ eine Parabel mit $p|_{[a,b]} \geq 0$ auf dem Intervall $[a,b] \subset \mathbb{R}$, $(a < b)$. Zeigen Sie:

$$A_P = A_T + \tfrac{4}{3} A_D$$

wobei

$$A_P := \int_a^b p(x)\,\mathrm{d}x$$

die Fläche unter der Parabel (grau), A_D die Fläche des Dreicks mit den Eckpunkten $(a, p(a))$, $(b, p(b))$ und $(\frac{a+b}{2}, p(\frac{a+b}{2}))$ (hellgrau) und A_T die Fläche des Trapezes mit den Eckpunkten $(a,0)$, $(b,0)$, $(a,p(a))$, $(b,p(b))$ (dunkelgrau) ist, vgl. Abbildung 1.

Warum ist damit die Aussage aus der Vorlesung – Beispiel (1.2) 1) (Fläche unter einer Parabel nach Archimedes) – gezeigt?

Hinweis: Verwenden Sie die Simpson-Regel. Die Fläche des Dreiecks ist als negativ zu werten, falls die Parabel nach oben geöffnet ist.

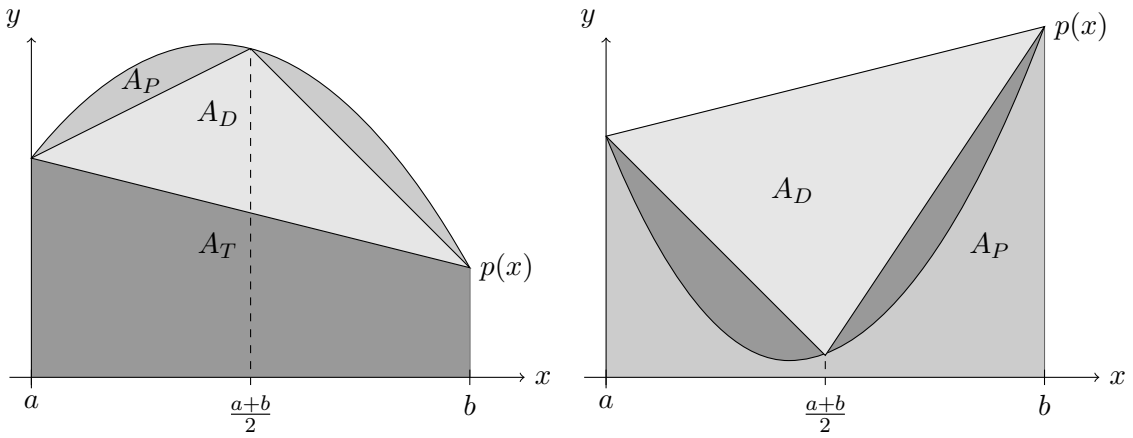

Abbildung 1:

Aufgabe 5: (6 Punkte)

Beweisen Sie, dass die Mittelpunktsregel optimal ist in dem Sinne, dass keine andere 1-stufige Quadraturformel Ordnung zwei oder mehr besitzt.

b.w.

Aufgabe 6: (6 Punkte)

Beweisen Sie: Falls die Knoten einer Quadraturformel $(b_i, c_i)_{i=1}^s$ paarweise verschieden und symmetrisch im Sinne der Definition (2.5) i) sind, d.h.

$$c_i = 1 - c_{s+1-i},\ i = 1, \ldots, s$$

erfüllen und ihre Ordnung $p \geq s$ ist, dann müssen auch die Gewichte symmetrisch sein, d.h.

$$b_i = b_{s+1-i},\ i = 1, \ldots, s$$

erfüllen.

Programmieraufgabe 2:

(a) Implementieren Sie (in PYTHON, MATLAB, ...) eine Funktion
`intApprox = numint(f, alpha, beta, N, b, c)`, welche für eine beliebige Quadraturformel $(b_i, c_i)_{i=1}^s$ eine gegebene Funktion $f : [\alpha, \beta] \to \mathbb{R}$, sowie eine Zerlegung

$$[\alpha, \beta] = [x_0, x_1] \cup [x_1, x_2] \cup \ldots \cup [x_{N-1}, x_N]$$

mit konstanter Gitterweite $h = x_{j+1} - x_j$, $j = 0, \ldots, N-1$ die Approximation

$$\int_\alpha^\beta f(x)dx \approx \sum_{j=1}^{N} h \sum_{i=1}^{s} b_i f(x_{j-1} + hc_i) \tag{1}$$

berechnet. Vermeiden Sie die Benutzung von `for`-Schleifen, d.h. implementieren Sie Ihre Funktion vektorisiert.

(b) Definieren Sie die Knoten und Gewichte der Rechtecksregel, Mittelpunktsregel, Trapezregel, Simpsonregel und die beiden zweistufigen Verfahren $((b_1, b_2), (c_1, c_2)) = ((\frac{3}{4}, \frac{1}{4}), (\frac{1}{3}, 1))$ und $((b_1, b_2), (c_1, c_2)) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2} - \frac{\sqrt{3}}{6}, \frac{1}{2} + \frac{\sqrt{3}}{6}))$.

Laden Sie sich den PYTHON-Modul `p2daten.py` von der Vorlesungshomepage herunter. Dieser definiert die Funktion `f` $= x \mapsto f(x)$, die Intervallgrenzen `alpha`, `beta` $\in \mathbb{R}$ und den Vergleichswert `exakt` $= \int_\alpha^\beta f(x)\,\mathrm{d}x$.

Falls Sie eine andere Programmiersprache verwenden, müssen Sie diesen Modul entsprechend übersetzen.

(c) Schreiben Sie ein Programm, um Ihr Verfahren wie folgt zu testen: Messen Sie den Approximationsfehler von (1) und die Zeit, die `numint` für die Verfahren aus (b) benötigt für `N` $= 10, 20, 50, 100, 200, 500, 1000$.

Tragen Sie in zwei doppelt-logarithmischen Diagrammen einmal Schrittweite gegen Fehler und einmal Zeit gegen Fehler auf. Denken Sie an Legenden und Beschriftungen.

Abgabe der Übungsaufgaben am Donnerstag, 4. Mai zu Beginn der Vorlesung.
Abgabe der Programmierübungen per E-Mail bis Freitag, 5. Mai, 23:59 Uhr an `num1@hhu.de` mit Betreff `PA# Gr#`, wobei `#` für die Nummer der Programmieraufgabe bzw. der Programmierübungsgruppe steht.