Technische Universität München
Institut für Informatik
Prof. Dr. Hans-Joachim Bungartz
Michael Lieb, M. Sc.
Dipl.-Inf. Gerrit Buse

WS 2009/2010

Numerisches Programmieren

4. Programmieraufgabe: Quadratur

Quadratur

Einfache Quadraturregeln

Bei den einfachen Quadratur-Regeln wird das Integrationsgebiet nicht zerlegt, die Regeln werden einmal auf das gesamte Integrationsgebiet angewandt. Im Rahmen dieser Programmieraufgabe sollen folgende einfache Regeln zur Integration der Funktion f im Integrationsgebiet $[a, b]$ implementiert werden:

- Rechtecksregel: Hierbei wird zur Bestimmung der Fläche unter der Funktion f der Funktionswert in der Mitte des Integrationsgebiets berechnet und mit der Breite $H = b - a$ des Integrationsgebiets multipliziert. Die berechnete Fläche entspricht also der Fläche des Rechtecks mit der Breite H und der Höhe $f(\frac{a+b}{2})$.
- Trapezregel: Für die Trapezregel werden die Funktionswerte am Rand des Integrationsgebiets benötigt. Die berechnete Fläche entspricht der des Trapezes mit der Breite $H = b - a$ und den Höhen $f(a)$ (linke Seite des Trapezes) bzw. $f(b)$ (rechte Seite des Trapezes).
- Keplersche Fassregel: Rechtecks- und Trapezregel entsprechen der Fläche unter dem konstanten bzw. linearen Interpolanten. Die Keplersche Fassregel berechnet die Fläche unter dem quadratischen Interpolanten. Die Stützstellen sind die Ränder und die Mitte des Integrationsgebiets.

Die *Newton-Cotes-Formeln* sind eine Verallgemeinerung der obigen Regeln für eine beliebige Anzahl an Stützstellen. Ab acht Stützstellen wird die Integration mit Hilfe dieser Formeln allerdings numerisch instabil.

Zusammengesetzte Quadraturregeln

Um dennoch ausreichend genau Integrationen komplexer Funktionen zu bestimmen, werden zusammegesetzte Regeln verwendet. Dabei wird das Integrationsgebiet in kleinere Gebiete unterteilt. Wir setzten zunächst voraus, dass das Integrationsgebiet in n gleich große Gebiete unterteilt wird. Das einfachste zu implementierende Verfahren ist die Anwendung der Rechtecksregel auf alle Teilgebiete und die anschließende Summation der Zwischenergebnisse.

Auch aus der Trapezregel und der Keplerschen Fassregel sollen zusammengesetzte Regeln gebaut werden. Man könnte auch hier einfach das Gebiet in n Teilgebiete aufteilen und auf jedes Gebiet die Trapezregel bzw. die Fassregel anwenden. Bei Funktionen, deren Auswertung einen hohen Rechenaufwand erfordert, ist diese Methode aber nicht sinnvoll. Wenn wir z.B. für die Trapezregel das Integrationsgebiet in n Teilgebiete aufteilen und auf jedes die Trapezregel anwenden, muss die zu integrierende Funktion $2n$ Mal ausgewertet werden, da die einfache Trapezregel zwei Funktionsauswertungen benötigt. Allerdings wird die Funktion nur an $n+1$ Stützstellen ausgewertet, d.h. sie wird an allen Stellen (außer denen am Rand des gesamten Integrationsgebiets) zwei Mal ausgewertet. Für Ihre Implementierung der Trapezsumme müssen Sie sicherstellen, dass die zu integrierende Funktion bei einer Unterteilung in n Gebiete genau $n+1$ Mal ausgewertet wird.

Dieselbe Anforderung gilt für Ihre Implementierung der Simpson-Summe (zusammengesetzte Fassregel), d.h. bei der Unterteilung in n Teilgebiete darf die Funktion f genau $n+1$ mal ausgewertet werden. Da die Simpson-Summe aber drei Stützpunkte benötigt, muss auf je zwei benachbarte Teilgebiete die Fassregel angewendet werden. Es muss daher eine gerade Anzahl an Teilgebieten verwendet werden.

Romberg-Quadratur (siehe Tutorübung - Blatt 7)

Wenn man von Rundungsfehlern absieht, werden die bisher behandelten zusammengesetzten Verfahren mit größerer Anzahl an Teilgebieten immer genauer. Wenn man die Breite h der einzelen Teilgebiete gegen Null gehen lässt, geht auch der Fehler gegen Null. Aufgrund des oftmals großen Rechenaufwands beim Auswerten der Funktion f lässt sich die Anzahl an Teilintervallen aber nicht beliebig vergrößern.

Die Idee der Romberg-Quadratur ist nun, zu verschiedenen h die Trapezsumme zu berechnen. Die verschiedenen h werden nun als Stützstellen und die zugehörigen Werte der Trapezsummen als Stützwerte eines (vgl. Tutroübung Blatt 7, Aufgabe 1) zu interpolierenden Polynoms (in h^2) verwendet. Eine Näherung für die Integration mit h gegen Null (d. h. "optimale"Integration) erhält man nun, wenn man das Interpolationspolynom an der Stelle Null ($h^2 = 0$) auswertet. Da diese Auswertestelle nicht innerhalb des Stützstellenintervals liegt, spricht man nicht von Interpolation, sondern von Extrapolation.

Adaptive Trapezsumme

Die bisher angesprochenen zusammengesetzten Verfahren zur Quadratur unterteilen das Gebiet in n gleich große Teile. In Bereichen, in denen die zu integrierende Funktion sehr glatt ist, sind nur wenige Unterteilungen nötig, wohingegen in Bereichen, in denen die Funktion nicht glatt ist, mehr Unterteilungen für eine gute Quadratur nötig sind. Daher soll eine weitere Methode implementiert werden, die zunächst die Trapezsumme mit einer gegebenen Zahl an Teilgebieten berechnet, dann aber an bestimmten Stellen weiter verfeinert. In Abb. 1 ist ein Teilgebiet von a bis b mit den zugehörigen Funktionswerten f_a und f_b abgebildet. Wenn nun die Differenz e zwischen dem Funktionswert in der Intervallmitte ($f(\frac{a+b}{2})$) und dem Mittel der Funktionswerte an den Intervallrändern ($\frac{f_a+f_b}{2}$) größer ist als ein gegebener Maximalwert, so soll weiter verfeinert werden. Da der Funktionswert in der Intervallmitte zum Prüfen dieser Bedingung ohnehin berechnet werden muss, wird er für einen letzten Ver-

feinerungsschritt verwendet. Auch bei dieser Implementierung darf zu jeder Stützstelle der Funktionswert nur einmal berechnet werden. Die Methode berechnet daher eine Trapezsumme mit n Teilintervallen unterschiedlicher Länge und muss dazu die gegebene Funktion $n+1$ Mal auswerten.

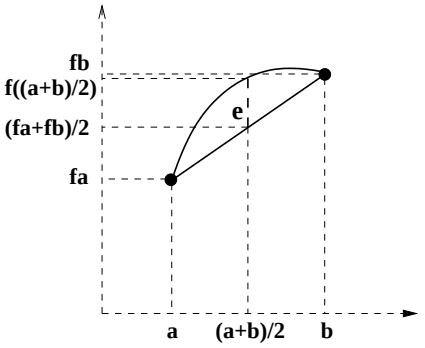

Abbildung 1: Abbruchkriterium

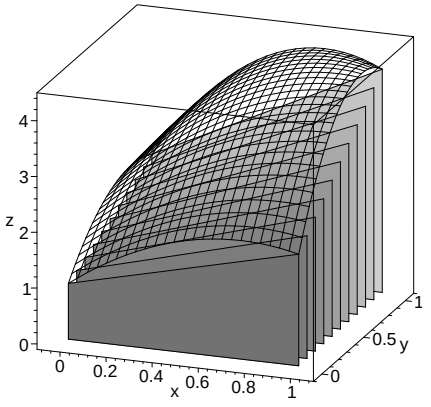

Abbildung 2: zweidimensionale Trapezregel

Zweidimensionale Quadratur

Bisher wurden nur Algorithmen zur eindimensionalen Quadratur implementiert, diese entsprechen der Berechnung der Fläche unter einer Funktion. Bei der zweidimensionalen Quadratur wird nun das Volumen unter einer Fläche berechnet. Bei allen zu implementierenden Verfahren werden nur rechteckige Gebiete verwendet. Zunächst soll die zweidimensionale Rechtecksregel implementiert werden. Dabei wird analog zur eindimensionalen Rechtecksregel der Funktionswert in der Mitte des Gebiets berechnet. Dieser Wert wird mit der Grundfläche des Gebiets multipliziert. Aus dieser einfachen Regel lässt sich auch im Zweidimensionalen eine zusammengesetzte Regel bauen. Man unterteilt das Gebiet in x-Richtung in n_x Teile und in y-Richtung in n_y Teile und erhält damit $n_x \cdot n_y$ Teilgebiete, auf die jeweils die Rechtecksregel angewandt wird.

Desweiteren soll die zweidimensionale Trapezregel implementiert werden, d.h. die Berechnung des Volumens unter einer Fläche. Als Stützstellen dienen hier die Eckpunkte des Integrationsgebiet $((x_a, y_a), (x_b, y_a), (x_a, y_b), (x_b, y_b))$. Zunächst wird die eindimensionale Trapezregel auf die Stützstellen (x_a, y_a) und (x_b, y_a) angewandt, womit man die Fläche A unter der Verbindungslinie der zugehörigen Stützpunkte erhält (siehe Abbildung 2). Daraufhin wird die eindimensionale Trapezregel auf die Stützstellen (x_a, y_b) und (x_b, y_b) angewandt, was die Fläche B unter der Verbindungslinie der zugehörigen Stützpunkte ergibt. Wenn nun zur Stützstelle y_a die erste Fläche A als Stützwert genommen wird und zur Stützstelle y_b die zweite Fläche B als Stützwert genommen wird, so kann durch erneutes Anwenden der Trapezregel das gesuchte Volumen berechnet werden. Auch aus der einfachen zweidimensionalen Trapezregel soll die zusammengesetzte zweidimensionale Trapezregel aufgebaut werden. Dies geschieht analog zur zweidimensionalen Rechtecksregel. Allerdings ist wie schon bei der eindimensionalen Trapezsumme zu beachten, dass die Funktion an jeder Stützstelle nur einmal ausgewertet wird.

Programmrahmen

Die Klasse *Funktion* erlaubt das Auswerten von Funktionen. Beim Instanziieren eines Objekts muss ein String übergeben werden, der die gewünschte Funktion in postorder-Notation enthält. Beispiele finden Sie im mitgelieferten Testprogramm und im Kommentar zur Klasse *Funktion*. Die Funktion kann von bis zu zwei Werten abhängen, die im übergebenen String x bzw. x und y benannt sein müssen. Der String kann außerdem neben Zahlen und den Zeichen für die Grundrechenarten noch weitere Operatoren enthalten. Details können Sie den Kommentaren entnehmen. Die Funktion wird mit der Methode *eval* ausgewertet. Je nach Funktion müssen dazu ein oder zwei double-Werte übergeben werden.

Alle zu implementierenden Methoden sind statische Methoden in der Klasse *Quadratur*. Das Testprogramm (*Test.java*) testet sämtliche Methoden mit einem einfachen Testfall. Beachten Sie, dass Sie Ihre Methoden auf jeden Fall mit weiteren Beispielen testen sollten!

Konkrete Aufgaben

Im Folgenden werden die zu implementierenden Methoden aufgelistet. Details zur Implementierung finden Sie jeweils in den Kommentaren zu den einzelnen Methoden. Testen Sie unbedingt jede einzelne Methode. Dazu können Sie z.B. die mitgelieferte Datei *Test.java* anpassen und erweitern. Zu implementieren sind:

- einfache Quadraturregeln 1D: *rechteck*, *trapez* und *simpson*
- zusammengesetzte Quadraturregeln 1D: *rechtecksumme*, *trapezsumme* und *simpsonsumme*
- Extrapolation: *romberg*
- adaptive Trapezsumme: *trapezsumme_adaptiv*
- einfache Quadraturregeln 2D: *rechteck2D* und *trapez2D*
- zusammengesetzte Quadraturregeln 2D: *rechtecksumme2D* und *trapezsumme2D*

Zur leichteren Implementierung der adaptiven Trapezsumme dürfen Sie weitere statische Methoden in die Klasse *Quadratur* aufnehmen.

Formalien

- Das Programmgerüst erhalten Sie auf den Webseiten zur Vorlesung.
- Ergänzen Sie das Programmgerüst bitte **nur an den dafür vorgegebenen Stellen!** Falls Sie die Struktur der Programme eigenmächtig verändern, können wir sie evtl. nicht mehr testen.
- Beseitigen Sie vor Abgabe Ihres Programms alle Ausgaben an die Konsole!
- Bitte reichen Sie Ihre Abgabe bis zum **1. Februar 2009, 12:00 Uhr** über das Web-Portal ein. Die Auswertung steht dann ab voraussichtlich 03. Februar, 12:00 Uhr zum Abruf bereit.