Dipl.-Math. Christian Himpe, Dipl.-Math. Felix Schindler WS 2013/2014

Praktikum:

Einführung in die Programmierung zur Numerik mit C++

Dienstag, 17.09.2013

Aufgabe 1 (Ableitung)
Schreiben Sie ein Programm `diffquot`, welches die Ableitung der Funktion $f(x) = x^2$ mit Hilfe des Vorwärtsdifferenzenquotienten

$$\frac{f(x+h) - f(x)}{h} = f'(x) + h.$$

berechnet. Da $f'(x) + h \approx f'(x)$ für kleines $h \in \mathbb{R}_+$, lässt sich der Vorwärtsdifferenzenquotient als Approximation für $f'(x)$ verwenden.
Gehen Sie folgendermaßen vor:

1. Schreiben Sie eine Funktion names `evaluate`, welche die Funktionswerte für $f(x) = x^2$ berechnet.
2. Schreiben Sie eine weitere Funktion `differentiate`, welche den Differenzenquotienten für festes $h > 0$ berechnet (unter Nutzung von `evaluate`).
3. Schreiben Sie ein Haupt-Programm, das als Kommandozeilen-Parameter die Intervallgrenzen a und b, die Anzahl N von Gitterpunkten und den Dateinamen einer Ausgabedatei übergeben bekommt. Die Schrittweite h wird auf $h = \frac{b-a}{N}$ gesetzt. D.h.

   $$a = x_0 < x_1 < ... < x_{N-1} < x_N = b, \quad \text{wobei } x_n = a + n \cdot h.$$

   Das Programm soll die Ableitung der Funktion (in x_n) mittles obigem Differenzenquotienten approximieren und eine Textdatei erzeugen, die zeilenweise die Ableitung zu den jeweiligen x-Werten enthält.
4. Plotten Sie die Ableitung sowie die Originalfunktion auf dem Intervall $[0, 3]$ mit $N = 300$ Unterteilungen mit Hilfe von `gnuplot`.
5. Tauschen sie $f(x) = x^2$ durch $f(x) = x^3 + x^2 + 2x + 1$ aus, in dem sie nur die `evaluate`-Methode verändern.

Aufgabe 2 (Ein einfacher ODE-Löser)
Schreiben Sie ein Programm `euler.cc`, welches die Anfangswertaufgabe

$$y'(x) = \frac{\sqrt{1 - y(x)^2}}{1 + x^2}, \quad y(0) = y_0 = 0$$

im Intervall $[0, 1]$ mit dem Eulerverfahren löst.

Gehen Sie folgendermaßen vor:

1. Schreiben Sie eine Funktion names `evaluate`, welche die Funktionswerte für
$$f(x,y) = \frac{\sqrt{1-y^2}}{1+x^2}$$
berechnet.
2. Schreiben Sie eine weitere Funktion `euler`, welche das Euler-Verfahren durchführt.
3. Schreiben Sie ein Haupt-Programm, das als Kommandozeilen-Parameter die Intervallgrenzen a und b, den Anfangswert y_0, die Anzahl der Gitterpunkte N, und den Dateinamen einer Ausgabedatei übergeben bekommt. Das Programm soll die Anfangswertaufgabe mit dem Eulerverfahren lösen und eine Textdatei erzeugen, die zeilenweise die Lösung zu den jeweiligen x-Werten enthält.
4. Plotten Sie die Lösung auf dem Intervall $[0,1]$ mit $N = 100$ (d.h. Schrittweite $h = 0.01$) mit Hilfe von `gnuplot`.
5. Ändern Sie die `evaluate`-Methode, um das Problem
$$\bar{y}'(x) = \sqrt{1-\bar{y}(x)^2}, \quad \bar{y}(0) = 0$$
im Intervall $[0,1]$ zu lösen. Vergleichen Sie das Ergebnis mit der exakten Lösung $\bar{y}(x) = \sin(x)$.
6. (Bonusaufgabe) Implmentieren Sie den zweistufigen Adams-Bashforth Löser (Beachten Sie das Sie für dieses Verfahren 2.ter Ordnung auch einen Startwert entsprechender Ordnung benötigen).

Zur Erinnerung: Das allgemeine Eulerverfahren wird durch
$$y_{n+1} = y_n + h \cdot f(x_n, y_n)$$
realisiert. In unserem Fall ist $f(x,y) = \frac{\sqrt{1-y^2}}{1+x^2}$ bzw. $f(x,\bar{y}) = \sqrt{1-\bar{y}^2}$.

Aufgabe 3 (ODE-Löser mit Hilfsfunktionen)

Schreiben Sie ein Programm `rk.cc`, welches die Anfangswertaufgabe
$$y'(x) = f(x, y(x)) = 1 + y^2(x), \quad y(0) = 0$$
im Intervall [0,1] mit dem Runge-Kutta-Verfahren vierter Ordnung löst. Testen sie Ihr Programm mit der Schrittweite h=0.01 und visualisieren Sie Ihre Lösung.

Zur Erinnerung: Das Runge-Kutta-Verfahren vierter Ordnung wird durch
$$y_{n+1} = y_n + \frac{h}{6}(f_1 + 2 \cdot f_2 + 2 \cdot f_3 + f_4)$$
realisiert, wobei
$$\begin{aligned} f_1 &= f(x_n, y_n), \\ f_2 &= f(x_n + \frac{h}{2}, y_n + \frac{h}{2} f_1), \\ f_3 &= f(x_n + \frac{h}{2}, y_n + \frac{h}{2} f_2), \\ f_4 &= f(x_n + h, y_n + h \cdot f_3). \end{aligned}$$

Gehen Sie hierbei wie bei Aufgabe 2 vor:

1. Schreiben Sie eine Funktion names `evaluate`, welche die Funktionswerte für
$$f(x, y) = 1 + y^2$$
berechnet.
2. Schreiben Sie eine weitere Funktion `rungekutta`, welche den Runge-Kutta-Schritt
$$y_{n+1} = y_n + \frac{h}{6}(f_1 + 2 \cdot f_2 + 2 \cdot f_3 + f_4)$$
durchführt.
3. Dies sind die einzigen Änderungen im Vergleich zu Aufgabe 2. Deshalb lässt sich das alte Programm leicht ergänzen, in dem man einfach die zusätzliche Methode `rungekutta` hinzufügt, `evaluate` ändert und sonst alles beim Alten lässt.
4. Schauen Sie sich die Lösung auf dem Intervall $[0, 1]$ mit $N = 100$ (d.h. Schrittweite $h = 0.01$) mit Hilfe von `gnuplot` an.
5. Lösen Sie das Problem
$$y'(x) = \frac{\sqrt{1 - y(x)^2}}{1 + x^2}, \quad y(0) = y_0 = 0$$
in $[0, 1]$ und vergleichen Sie die Lösung mit der aus dem Euler-Verfahren.

Aufgabe 4 (Komponenten eines Dateipfades)

1. Schreiben Sie ein Programm, das bei einem via Kommandozeilenargument gegebenen Pfad wie z.B:
`/this/is/my/special/path/to/a/file/myfile.txt`
die Komponenten Pfad (hier: `/this/is/my/special/path/to/a/file/`), Dateiname (hier: `myfile`) und Erweiterung (hier: `.txt`) sollen extrahiert und ausgegeben werden.
2. Beachten Sie, dass nicht jeder Pfad mit `/` beginnt und auch nicht jede Datei eine Endung hat.
3. Falls kein Kommandozeilenargument vorhanden ist soll eine interaktive Abfrage erfolgen.
4. Welche Funktion aus den C-Bibliotheken kann hier hilfreich sein?
5. Welche Signatur (Rückgabewert und Argumente) hätte eine Funktion die diese Komponenten zur Verfügung stellen soll?