Institut für Angewandte und Numerische Mathematik
Prof. Dr. Tobias Jahnke • Dipl.-Math. oec. Marcel Mikl

Numerische Mathematik 1

Wintersemester 2014/15

Programmierblatt 2

Aufgabe 5 (Knoten und Gewichte der Gauß-Quadratur)

Schreiben Sie eine MATLAB-Funktion

```
[b, c] = gaussqf(s),
```

welche die Knoten und Gewichte der Gauß-Quadraturformel ohne Gewichtsfunktion mit s Knoten berechnet. Gehen Sie dabei so vor, wie in Kapitel 1.8 der Vorlesung beschrieben. Verwenden Sie dazu die Rekursionsformel

$$\gamma_{k+1} P_{k+1}(x) = x P_k(x) - \gamma_k P_{k-1}(x), \quad \gamma_k = \frac{k}{\sqrt{4k^2-1}}, \quad k = 1, 2, \ldots$$

mit

$$P_{-1} = 0, \quad P_0 = \frac{1}{\sqrt{2}} \quad \text{und} \quad \gamma_0 = \sqrt{2}$$

für die Legendre-Polynome. Diese normiert die Legendre-Polynome so, dass $\langle P_k, P_k \rangle = 1$ gilt und sich damit die Rekursionsformel entsprechend vereinfacht.

Vorgehensweise:

- Stellen Sie eine geeignete symmetrische Tridiagonalmatrix auf.
- Berechnen Sie die Eigenwerte und Eigenvektoren der Tridiagonalmatrix.
- Verwenden Sie die aus der Vorlesung bekannten Zusammenhänge zwischen den Eigenwerten und Eigenvektoren und den Knoten und Gewichten der Gauß-Quadraturformel.

Überprüfen Sie die Ergebnisse Ihrer Funktion mithilfe der Funktion `p = quadord(b,c)` aus Aufgabe 4.

Hilfreiche Befehle: `diag`, `eig`

Aufgabe 6 (Dividierte Differenzen)

(a) Schreiben Sie eine MATLAB-Funktion

```
y = divdif(x,y),
```

die für gegebene Punkte $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, gespeichert als Vektoren `x` und `y`, die dividierten Differenzen berechnet und sie wieder in `y` abspeichert.

Vorgehensweise:
Überlegen Sie sich, wie die Berechnung in den zwei `for`-Schleifen

```
for j=1:n
   for i=n:-1:j+1
      ⋮
   end
end
```

durchgeführt werden kann.

(b) Schreiben Sie eine MATLAB-Funktion

```
z = polyval(x,y,t),
```

die das Interpolationspolynom zu den Daten $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, gespeichert als Vektoren `x` und `y`, an der Stelle t auswertet. Verwenden Sie innerhalb der Funktion die MATLAB-Funktion `divdif` zur Berechnung der dividierten Differenzen und das *Hornerschema für Newton'sche Interpolationspolynome*.

(c) Testen Sie Ihre Programme, indem Sie zu $x_i = 1+0.2i$, $i = 1, \ldots, 5$ und den Werten $y_i = \log(x_i)$ das Interpolationspolynom an den Stellen $t_i = 1.1, 1.3, 1.5, 1.7, 1.9$ auswerten. Geben Sie dazu t_i, $p(t_i)$ und $\log(t_i)$ aus und vergleichen Sie.

Die Aufgaben werden am **Donnerstag, den 20. November 2014, 15:45 Uhr** und am **Montag, den 24. November 2014 , 09:45 Uhr** in den Programmiertutorien besprochen.

Homepage:

Unter http://www.math.kit.edu/ianm3/lehre/numa12014w/ erreichen Sie die Homepage zur Vorlesung. Dort finden Sie neben den aktuellen Übungsblättern auch alle Informationen zum Vorlesungsbetrieb.