



## Numerik

Wintersemester 2019/20

Tübingen, 13.11.2019

### Übungsblatt 5

**Problem 1.** Bestimmen Sie die Anzahl der Rechenoperationen (jeweils die Anzahl der Additionen bzw. Multiplikationen), die für die Berechnung eines Interpolationspolynoms in Lagrange-Darstellung nötig sind.

**Problem 2.** Bestimmen Sie die Anzahl der Rechenoperationen beim dividierten Differenzen-Verfahren mit Newton-Polynomen. Vergleichen Sie diese mit der aus **Problem 1**. Was ist — abgesehen von weniger Rechenschritten — der Vorteil gegenüber dem Lagrange-Ansatz?

**Problem 3.** Gegeben sei die folgende Wertetabelle für die Secans Hyperbolicus Funktion  $f(x) = \operatorname{sech}(x)$ :

$x_i$	0.9	0.95	1.00	1.05	1.10
$f(x_i)$	0.697795	0.672845	0.648054	0.623521	0.599334

Bestimmen Sie durch Extrapolation des zentrierten Differenzenquotienten

$$a(h) := \frac{f(x+h) - f(x-h)}{2h}$$

eine Näherung zum Ableitungswert  $f'(1)$ . Rechnen Sie dabei mit 6 signifikanten Stellen.

**Hinweis:** Da  $f$  analytisch ist, gilt

$$a(h) = f'(x) + \sum_{i=1}^{\infty} \frac{f^{(2i+1)}(x)}{(2i)!} h^{2i}$$

#### Programmieraufgabe 2: (Polynominterpolation).

Es sei die Funktion  $f: \mathbb{R} \rightarrow \mathbb{R}$  durch

$$f(x) = \frac{1}{1 + 25x^2}$$

gegeben.

- a) Schreiben Sie eine MATLAB-Funktion `NewtonInter(x, y, u)`, welche für zwei übergebene Vektoren  $x, y \in \mathbb{R}^{n+1}$ , die die Punkte  $(x_i, y_i)$  für  $i = 0, \dots, n$ , beinhalten und einen übergebenen Vektor  $u \in \mathbb{R}^m$  die Koeffizienten  $c_i$  des Newton-Interpolationspolynoms  $p$ , sowie simultan die

Funktionswerte  $v_j = p(u_j)$  für  $j = 1, \dots, m$  mit Hilfe der dividierten Differenzen berechnet (ohne die Verwendung der beiden Matlab-Befehle `polyfit` und `polyval`) und in geeigneten Vektoren  $c \in \mathbb{R}^{n+1}$  bzw.  $v \in \mathbb{R}^m$  zurückgibt. Sie dürfen hierbei annehmen, dass die Stützstellen  $x_i$  paarweise verschieden sind. Die MATLAB-Funktion soll dabei folgende Gestalt haben:

```
1     function [v,c] = NewtonInter(x,y,u)
2     ...
3     end
```

b) Implementieren Sie obige Funktion  $f$ , die zu  $x$  den Wert  $f(x)$  liefert als MATLAB-Funktion. Die Funktion kann dann mittels `f(x)` aufgerufen werden und soll dabei folgende Gestalt haben:

```
1     function [y] = f(x)
2     ...
3     end
```

c) Schreiben Sie das MATLAB-Skript `main.m`, welches unter Verwendung der Funktion `NewtonInter` folgende Punkte realisiert:

Interpolieren Sie die Funktion  $f$  im Intervall  $[-1, 1]$  unter der Verwendung von Stützstellen  $x_i = \cos\left(\frac{(2i+1)\pi}{2n+2}\right)$  mit  $i = 0, \dots, n$  für  $n = 4, 8, 12$ . Plotten Sie die resultierenden Interpolationspolynome zusammen mit  $f$  in ein geeignetes Schaubild. Plotten Sie den Interpolationsfehler  $|p(x) - f(x)|$  für alle  $x \in [-1, 1]$  im Falle  $n = 12$  in ein weiteres Schaubild. Verwenden Sie hierbei die MATLAB-Befehle `figure(1)` und `figure(2)` um unterschiedliche Schaubilder zu initialisieren.

**Besprechung der Aufgaben in den Übungsgruppen am 19.11.2019. Abgabe von Programmieraufgabe 2 bis spätestens 27.11.2019 per Mail an: " [progtutor@na.uni-tuebingen.de](mailto:progtutor@na.uni-tuebingen.de) ".**