Numerische Differentiation und Integration

Aufgabe 1 Differentiation:

(a)

- Programmieren Sie eine Matlab File zur numerischen Differentiation einer beliebigen analytischen Funktion basierend auf der Definition des Differenzenquotienten

$$\frac{df(x)}{dx} = \lim\Big|_{\Delta x \to 0} \left(\frac{f(x+\Delta x) - f(x)}{\Delta x} \right) \approx \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}.$$

Funktionsaufruf:

```
[xwerte,ableitung]=diffquot(fhandle,a,b,N
);
```

- Die Definition der zu differenzierenden Funktion f(x) soll extern erfolgen. Nutzen Sie hierbei die in Matlab vorgesehene Übergabe von Funktionen.
- Die im Intervall [a,b] gefundene Ableitung soll bei jedem Funktionsaufruf graphisch dargestellt werden.

(b)

- Überprüfen Sie am Beispiel des Sinus im Intervall von 0 bis 2π die Richtigkeit der Berechnung in Abhängigkeit von der Anzahl der Diskretisierungspunkte N.
- Stellen Sie dazu die numerisch gefundene Ableitung mit der analytischen für verschiedene N graphisch dar, indem Sie ein weiteres Programm schreiben, welches auf die in Aufgabenteil (a) erstellte Ableitungsfunktion zugreift.

```
konvergenz(fhandle,a,b,Nstart,Nende);
```

Hilfreiche Funktionen:

- `linspace(a,b,N)` *erzeugt einen Vektor mit N äquidistanten Einträgen im Intervall [a,b]*
- `fhandle=@(x) sin(x)` *verknüpft den Ausdruck fhandle mit einer Funktion in Abhängigkeit von x, hier dem Sinus von x*
- `plot(x,y)` *stellt y Werte über den übergebenen x Werten graphisch dar*

Aufgabe 2 Integration:

(a)

- Programmieren Sie eine Funktion zur numerischen Integration einer beliebigen analytischen Funktion mittels der Rechteck-Regel.

Funktionsaufruf:

```
[A] = intrect(fhandle,a,b,N);
```

$$\int_a^b f(x)dx = F(b) - F(a) = \lim\Big|_{\Delta x_i \to 0} \left(\sum_{i=1}^{N-1} \Delta x_i f(\xi_i) \right) ,$$

$$\Delta x_i = [x_i, x_{i+1}] \ , \ \xi_i \in \Delta x_i \ , \ x_1 = a \ , \ x_N = b \ .$$

- Ausgabeparameter A ist die Fläche unter der Funktion f(x) im Intervall [a,b]

(b)

- Ändern Sie das Programm aus Aufgabenteil (a) dahin gehend ab, dass die Stammfunktion F(x) von f(x) im Intervall [a,b] ausgegeben und graphisch dargestellt wird.

Aufgabe 3 Vergleich:

(a)

- Erstellen Sie ein Programm, welches die numerische Differentiation aus Aufgabe 1 und die numerische Integration aus Aufgabe 2 verknüpft. Resultat dieser Berechnung ist die ursprüngliche Funktion f(x).

- Stellen Sie das Resultat für den Spezialfall einer Gauß'sche Glockenkurve im Intervall [-10,10] dar für festes N dar.

$$f(x) = e^{-x^2} \quad , \quad x \in [-5,5] \ .$$

- Achten Sie dabei darauf, das nach der ersten Operation (Int. oder Diff.) die x Werte und die zugehörigen Funktionswerte (Stammfkt. oder Ableitung) als Daten vorliegen. Die zweite Operation muss dem zu folge mit diesen numerisch bestimmten Daten funktionieren, was einer Modifikation der Programme in den Aufgaben 1 und 2 bedarf. Mit welcher Operation muss man beginnen?

Funktionsaufruf:

```
[x,f] = intdiff(fhandle,a,b,N);
```

(b)*

- Um die Konvergenz der Integration und Differentiation getrennt betrachten zu können, soll für die Funktion aus (a) eine Konvergenzbetrachtung durchgeführt werden.
- Als Benchmarkparameter dienen sowohl analytisch als auch numerisch bekannte Größen. Für die Integration ist dies die FWHM (full width half maximum) Fläche unter der Funktion, welche sich im Intervall $\left[-\sqrt{\ln(0.5)}\,;-\sqrt{\ln(0.5)}\right]$ zu $A_{FWHM} = 1.3488$ ergibt.
- Für die Ableitung ist dies die Summe der quadratischen Abweichungen von der analytischen Ableitung im gleichen Intervall (Standardabweichung).

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} \left(f_{analyt.}'(x_i) - f_{num.}'(x_i) \right)^2}$$

Funktionenaufruf:

```
benchmark(a,b,fhandle,N,intbm,diffbm);
```

Hausaufgabe

Zahlendarstellung:

Programmieren Sie eine Funktion zur Konvertierung einer beliebigen römischen Zahl in Ihre Binärdarstellung. Funktionsaufruf:

```
x_bin=rom2bin(x_rom)
```

% x_bin=rom2bin(x_rom)
% Konvertierung einer römischen Zahl
% in ihre binäre Darstellung
% x_bin <= Binärdarstellung
% x_rom => Zeichenkette der römischen Zahl

Funktionen zur Verarbeitung von Zeichenketten (Matlab Hilfe)

– **strcmp** determines if two strings are identical
– **strncmp** determines if the first n characters of two strings are identical
– **findstr** returns the starting position of a substring within a longer string
– **strcat** Concatenate strings