Einführung in die Numerik
5. Programmierübung

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Mathematik
Prof. Dr. Herbert Egger, Dr. Mirjam Walloth, Thomas Kugler

WS 2015/16
13.1.2015

Programmierübung

Aufgabe P1

bisection.m: Gesucht ist eine Nullstelle der Funktion $f : \mathbb{R} \to \mathbb{R}$. Implementieren Sie eine Routine

```
function [x,xk]=bisection(f,a,b,tol)
```

welche die Nullstelle x mit Toleranz tol berechnet. Hierbei sei f eine stetige Funktion welche mit $f(x)$ ausgewertet werden kann. Weiters gelte $a < b$ und $f(a)f(b) < 0$. Der Rückgabevektor `xk` beinhaltet zur späteren Analyse die k-ten Approximationen an die Nullstelle.

regulaFalsi.m: Anstelle von $x_k = (a_k + b_k)/2$, wie im Bisektionsverfahren, wird der nächste Kandidat für die Nullstelle durch $s_k(x_k) = 0$ bestimmt, wobei s_k die lineare Interpolierende zu f an den Stellen a_k und b_k (=Sekante) ist. Implementieren Sie eine entsprechende Funktion

```
function [x,xk]=regulaFalsi(f,a,b,tol,maxit),
```

indem Sie das Bisektionsverfahren geeignet anpassen. Der Rückgabevektor ck beinhaltet wieder die Approximationen an die Nullstelle in den einzelnen Schritten.

testp1.m:

(a) Berechnen Sie mit obigen Algorithmen ausgehend von $[a_0, b_0] = [1, 2]$ die Nullstelle x^* von $f(x) = \cos(x)$ numerisch, sodass $|f(x)| \leq tol = 10^{-8}$ garantiert werden kann. Begründen Sie, warum somit auch $|x - x^*| < c \cdot tol$ erwartet werden kann.

(b) Tragen Sie den Iterationsindex k und die Fehler $e_k = |x_k - x^*|$ in einer Tabelle auf.

(c) Vergleichen Sie die Konvergenz des Fehlers mit der des Bisektions- und des Newton-verfahrens. Welche Aussage würden Sie über die Konvergenz der Regula-Falsi-Methode machen?

Aufgabe P2

newton1d.m: Implementieren Sie eine Routine

```
function [x,xk]=newton1d(f,df,x0,tol,maxit)
```

zur numerischen Berechnung der Nullstelle $f(x) = 0$ mit dem Newtonverfahren. Hierbei soll $f(x)$ den Funktionswert und $df(x)$ den Wert der Ableitung an der Stelle x zurückliefern. Der Vektor `xk` enthält zu Analysezwecken alle Iterierten des Newtonverfahrens. Die Iteration soll abgebrochen werden, sobald $|x^{k+1} - x^k| < tol$ ist, oder die maximale Iterationsanzahl erreicht ist.

secant.m: Um die Berechnung der Ableitung zu vermeiden, betrachten wir noch das Sekantenverfahren. Dabei wird die Ableitung $f'(x^k)$ im Newtonverfahren durch die Ableitung der Sekanten $s_k'(x^k) = (f(x^k) - f(x^{k-1}))/(x^k - x^{k-1})$ ersetzt. Implementieren Sie eine entsprechend Routine

```
function [x,xk]=secant(f,x0,tol,maxit)
```

zur numerischen Berechnung der Nullstelle $f(x) = 0$ mit dem Sekantenverfahren. Für den ersten Schritt kann man z.B. $x^{-1} := x^0 + 1e - 6$ setzen. Der Vektor `xk` enthält zu Analysezwecken alle Iterierten des Verfahrens. Die Iteration soll wieder abgebrochen werden, sobald $|x^{k+1} - x^k| < tol$ ist, oder die maximale Iterationsanzahl erreicht ist.

testp2.m: Wiederholen Sie die Tests aus Aufgabe P1 mit dem Sekanten und dem Newtonverfahren mit einem geeigneten Startwert x^0. Was können sie über die Konvergenzgeschwindigkeit der Verfahren sagen? Wie passt das zu den theoretischen Aussagen?

Aufgabe P3
newton.m: Implementieren Sie eine Routine

```
function [x,xk]=newton(F,DF,x0,tol,maxit)
```

zur numerischen Lösung des nichtlinearen Gleichungssystems $F(x) = 0$ mit dem Newtonverfahren. Hierbei soll $F(x)$ den Funktionswert (Vektor) und $DF(x)$ die Jacobimatrix an der Stelle x zurückliefern. Der Spalten der Matrix `xk` enthalten zu Analysezwecken alle Iterierten des Newtonverfahrens. Die Iteration soll abgebrochen werden, sobald $\|x^{k+1} - x^k\| < tol$ ist, oder die maximale Iterationsanzahl erreicht ist.
testp3.m: Lösen Sie mithilfe des Verfahrens das nichtlineare Gleichungssystem aus Beispiel 9.20 im Skriptum mit $tol = 10^{-10}$. Erstellen Sie eine Tabelle mit dem Iterationsindex k und den Fehlern $e_k = \|x^* - x^k\|$. Erklären Sie das Konvergenzverhalten.

Aufgabe P4
banach.m: Implementieren Sie eine Routine

```
function [x,xk]=banach(phi,x0,tol,maxit)
```

zur numerischen Lösung der Fixpunktgleichung $\phi(x) = 0$ mit der Banach'schen Fixpunktiteration. Hierbei soll $\phi(x)$ den Funktionswert der vektorwertigen Abbildung ϕ zurücklieren. Der Spalten der Matrix `xk` enthalten zu Analysezwecken alle Iterierten des Verfahrens. Die Iteration soll abgebrochen werden, sobald $\|x^{k+1} - x^k\| < tol$ ist, oder die maximale Iterationsanzahl erreicht ist.
testp4.m: Lösen Sie mithilfe des Verfahrens das nichtlineare Gleichungssystem aus Beispiel 9.10 im Skriptum mit $tol = 10^{-10}$. Erstellen Sie eine Tabelle mit dem Iterationsindex k und den Fehlern $e_k = \|x^* - x^k\|$. Erklären Sie das Konvergenzverhalten.

Aufgabe P5
gaussNewton.m: Implementieren Sie eine Routine

```
function [x,xk]=gaussNewton(F,DF,x0,tol,maxit)
```

zur numerischen Lösung des nichtlinearen Ausgleichsproblems $f(x) = \|F(x)\|^2 \to \min$ mit $F : \mathbb{R}^n \to \mathbb{R}^m$. Hierbei soll $F(x)$ den Funktionswert (Vektor) und $DF(x)$ die Jacobimatrix an der Stelle x zurückliefern. Der Spalten der Matrix `xk` enthalten zu Analysezwecken alle Iterierten des Verfahrens. Die Iteration soll abgebrochen werden, sobald $\|x^{k+1} - x^k\| < tol$ ist, oder die maximale Iterationsanzahl erreicht ist.
testp5.m: Verwenden Sie das Verfahren zur numerischen Lösung des Ausgleichsproblems aus Beispiel 11.3 im Skriptum. Geben Sie den Iterationsindex k und den Fehler $e^k = |a^* - a^k|$ tabellarisch an und erklären Sie das Konvergenzverhalten. Plotten Sie die Fehlerquadratfunktion $f(x)$ als blaue Linie und tragen Sie die Näherungen $(x^k, f(x^k))$ als rote Punkte ein.

Organisatorische Hinweise:

a) Legen Sie ein Verzeichnis `pp5` an und speichern Sie alle Funktionen und Skripte für diese Übung darin ab. Der Inhalt des Verzeichnisses sollte hier also sein:

```
testp1.m bisection.m regulaFalsi.m
testp2.m newton1d.m  secant.m
testp3.m newton.m
testp4.m banach.m
testp5.m gaussNewton.m
```

b) Zippen Sie das Verzeichnis; in Linux: `zip -r pp5.zip pp5` von außerhalb des Verzeichnisses. In Windows kann WinZip oder ein ähnliches Tool verwendet werden.

c) Überprüfen Sie, dass alle Skripte und Funktionen lauffähig sind!

d) Kommentieren Sie die Funktionen geeignet. Insbesondere sollten Eingabe und Ausgabe Parameter beschrieben werden. `help <funktionsname>` sollte Information über die Routinen liefern.

e) Antworten auf die Fragen sollten in den `testp*.m` files als Kommentare hinterlegt werden.