

Übungen zur Vorlesung Einführung in das Programmieren für TM

Serie 7

Die Aufgaben mit Stern (*) sind bis zur Übung in der kommenden Woche vorzubereiten. Kopieren Sie die Source-Codes vor der Übung auf Ihren Account auf der `lva.student.tuwien.ac.at` und überprüfen Sie, ob diese von `matlab` auf der `lva.student.tuwien.ac.at` interpretiert werden können. In den folgenden Aufgaben sollen **Arithmetik**, **Verzweigungen** und **Funktionen** geübt werden.

Aufgabe 7.1*. Matlab stellt Ihnen eine umfangreiche Bibliothek mit zahlreichen nützlichen Funktionen zur Verfügung. Zu jeder Funktion erhalten Sie mit `help funktionsname` eine ausführliche Beschreibung. Erklären Sie die Verwendung von `find` und `max`. Was sind die möglichen Parameter? Was sind die möglichen Rückgabewerte? Überlegen Sie sich einfache Beispiele an Hand derer Sie die Funktionen illustrativ erklären können und halten Sie diese in einem einfachen Skript fest.

Aufgabe 7.2*. Schreiben Sie eine Funktion `maxcompare`, die für zwei gegebene Vektoren $a, b \in \mathbb{R}^n$ zählt wie oft das Maximum $M = \max\{a_i, b_i \mid i = 1, \dots, n\}$ im Vektor a und b an der gleichen Stelle vorkommt. Zum Beispiel soll die Funktion für die Vektoren $a = (1.1, 4, 2e - 4, 4, 4, 3, 4, -1.5)$ und $b = (2.2, 4, 4, 2e - 5, 4, -1, 2.7, 4)$ den Wert 2 zurückgeben, da das Maximum $M = 4$ in beiden Vektoren an zwei Stellen nämlich $a_2 = b_2 = a_5 = b_5 = M$, gleichermaßen vorkommt. Speichern Sie den Source-Code unter `maxcompare.m` in das Verzeichnis `serie07`.

Aufgabe 7.3*. Was macht folgende Funktion?

```
function [y,z] = f(x,y,z)
if (x==1)
    if (y>z)
        tmp=y;
        y=z;
        z=tmp;
    end
else
    if (z>y)
        tmp=y;
        y=z;
        z=tmp;
    end
end
```

Wie könnte man eine Funktion programmieren, die dasselbe leistet, aber nur eine Verzweigung verwendet?

Aufgabe 7.4*. Schreiben Sie eine Funktion `pnorm`, die für $p \in [1, \infty)$ die ℓ_p -Norm

$$\|x\|_p := \left(\sum_{j=1}^n |x_j|^p \right)^{1/p}$$

eines Vektors $x \in \mathbb{R}^n$ berechnet. Anstatt Schleifen soll zur Summation der Befehl `sum` verwendet werden. Speichern Sie den Source-Code unter `pnorm.m` in das Verzeichnis `serie07`.

Aufgabe 7.5. Schreiben Sie eine Funktion `cut`, die zu gegebenem $k \in \mathbb{N}$ aus einem Vektor $x \in \mathbb{R}^n$ alle Einträge x_j mit $|x_j| \geq k$ streicht. Anstatt Schleifen soll der Befehl `find` verwendet werden.

Aufgabe 7.6. Die Summe $r = p + q$ zweier Polynome p, q ist wieder ein Polynom. Schreiben Sie eine Funktion `addPolynomials`, die die Summe r berechnet. Dabei sollen $p(x) = \sum_{k=1}^m a_k x^{k-1}$ und $q(x) = \sum_{k=1}^n b_k x^{k-1}$ in Form der Zeilenvektoren $a \in \mathbb{R}^m$ und $b \in \mathbb{R}^n$ ihrer Koeffizienten gespeichert werden. Realisieren Sie die Funktion ohne Verwendung von Schleifen mithilfe der MATLAB-Arithmetik.

Aufgabe 7.7. Schreiben Sie eine Funktion `evalPolynomial`, die den Funktionswert $p(x) = \sum_{k=1}^n a_k x^{k-1}$ zurückgibt. Dabei werde das Polynom in einem Zeilenvektor $a \in \mathbb{R}^n$ der Länge n gespeichert. Falls x ein Spaltenvektor der Länge m ist, soll $p(x)$ ebenfalls ein Spaltenvektor der Länge m sein. Realisieren Sie die Funktion wieder unter Vermeidung von Schleifen.

Aufgabe 7.8. Schreiben Sie eine Funktion `differentiatePolynomial`, die den Koeffizientenvektor der Ableitung $p'(x)$ des Polynoms $p(x) = \sum_{k=1}^n a_k x^{k-1}$ zurückgibt. Realisieren Sie die Funktion wieder ohne Verwendung von Schleifen.

Aufgabe 7.9. Schreiben Sie eine Funktion `skalarprodukt`, die das Skalarprodukt zweier Vektoren $x, y \in \mathbb{R}^n$ berechnet, ohne Schleifen zu verwenden. Dabei dürfen x und y Spalten- oder Zeilenvektoren sein, die ggf. mittels `reshape` auf passende Form gebracht werden.

Aufgabe 7.10. Schreiben Sie eine Funktion `minabs`, die von zwei Werten $x, y \in \mathbb{R}$ denjenigen zurückliefert, dessen Absolutbetrag kleiner ist. Der Absolutbetrag wird in Matlab durch `abs` gegeben. Realisieren Sie die Funktion mit und ohne die Verwendung von `min`.