

Übungen zur Vorlesung Einführung in das Programmieren für TM

Serie 5

Die Aufgaben mit Stern (*) sind bis zur nächsten Übung vorzubereiten und werden dort abgeprüft. Die übrigen Aufgaben dienen nur Ihrer Übung und mir als zusätzliche Grundlage für den Prüfungsstoff in den schriftlichen Tests. Kopieren Sie bitte den Source-Code in ein Unterverzeichnis `serie05` Ihres Home-Verzeichnisses. Überprüfen Sie bitte vor der Übung, ob Ihre Source-Codes mit dem `gcc` kompiliert werden können. In den folgenden Aufgaben sollen neben Schleifen auch **Call by Reference** und elementare **dynamische Vektoren** geübt werden.

Aufgabe 41*. Man schreibe eine Funktion `transpose`, die zu einer spaltenweise gespeicherten Matrix $A \in \mathbb{R}^{n \times n}$ die transponierte Matrix $A^T \in \mathbb{R}^{n \times n}$ berechnet. Dabei sind die Einträge von A^T gerade durch $(A^T)_{jk} = A_{kj}$ definiert. Die Matrix A^T soll ebenfalls spaltenweise gespeichert werden, und die Einträge der Matrix A sollen mit den Einträgen der Matrix A^T überschrieben werden. Die Dimension $n \in \mathbb{N}$ darf eine Konstante im Hauptprogramm sein. Die Funktion `transpose` ist aber für beliebige Dimension zu programmieren. Im Hauptprogramm soll A eingelesen und A sowie A^T ausgegeben werden. Speichern Sie den Source-Code unter `transpose` ins Verzeichnis `serie05`.

Aufgabe 42*. Für eine Matrix $A \in \mathbb{R}^{m \times n}$ ist die Spaltensummennorm durch

$$\|A\| = \max_{k=1, \dots, n} \sum_{j=1}^m |A_{jk}|$$

gegeben. Schreiben Sie eine Funktion `spaltensummennorm`, die die Spaltensummennorm einer Matrix A berechnet. Dabei sei die Matrix A spaltenweise gespeichert. Schreiben Sie ein aufrufendes Hauptprogramm, in dem A eingelesen und $\|A\|$ ausgegeben wird. Die Dimensionen $m, n \in \mathbb{N}$ der Matrix dürfen Konstanten im Hauptprogramm, müssen aber Parameter der Funktion `spaltensummennorm` sein. Speichern Sie den Source-Code unter `spaltensummennorm.c` ins Verzeichnis `serie05`.

Aufgabe 43*. Man schreibe eine Funktion `minmaxmean`, die von einem gegebenem Vektor $x \in \mathbb{R}^n$ das Minimum $\min_{j=1}^n x_j$, das Maximum $\max_{j=1}^n x_j$ sowie den Mittelwert $\frac{1}{n} \sum_{j=1}^n x_j$ berechnet und zurückgibt. Ferner schreibe man ein aufrufendes Hauptprogramm, das $n \in \mathbb{N}$ sowie den dynamischen Vektor $x \in \mathbb{R}^n$ einliest und Minimum, Maximum und Mittelwert von x ausgibt. Da eine C-Funktion maximal einen elementaren Datentyp als Return-Wert zurückliefern kann, realisiere man die Rückgabe mittels Call by Reference (Pointer!). Den Source-Code speichere man unter `minmaxmean.c` ins Verzeichnis `serie05`.

Aufgabe 44*. Man implementiere eine Funktion `eratosthenes`, die das *Sieb des Eratosthenes* realisiert. Dies ist ein Algorithmus, mit dem alle Primzahlen bis zu einer bestimmten Zahl `nmax`

errechnet werden können (benannt nach dem griechischen Mathematiker Eratosthenes). Der Algorithmus sieht folgendermaßen aus:

- Man legt eine Liste (Vektor) `prim = (2, ..., nmax) ∈ ℕnmax-1` an.
- Man streicht aus der Liste alle Vielfachen der ersten Zahl (also der Zahl 2).
- Wähle, solange es noch höhere Zahlen gibt, die nächsthöhere nicht durchgestrichene Zahl und streiche alle ihre Vielfachen.

Der Rückgabvektor `prim` soll am Ende (gekürzt auf minimale Länge) alle Primzahlen $\leq nmax$ enthalten (Sie müssen zusätzlich die Länge des Rückgabvektors zurückgeben!). Realisieren Sie das Streichen geeignet, z.B. indem Sie die entsprechenden Einträge auf 0 setzen. Speichern Sie den Source-Code unter `eratosthenes.c` ins Verzeichnis `serie05`.

Aufgabe 45. Man schreibe eine Funktion `maxcount`, die von einem Vektor $x \in \mathbb{R}^n$ das Maximum zurückliefert und die Anzahl, wie oft dieses im Vektor vorkommt. Ferner schreibe man ein aufrufendes Hauptprogramm, das die Länge $n \in \mathbb{N}$ und den Vektor $x \in \mathbb{R}^n$ einliest und das Ergebnis der Funktion ausgibt. Den Source-Code speichere man unter `maxcount.c` ins Verzeichnis `serie05`.

Aufgabe 46. Man schreibe eine verbesserte Variante von Bubble-Sort, bei dem man sich die Stelle des letzten Tausches merkt. Ab dieser Stelle müssen die Daten ja bereits sortiert sein. Der nächste Durchlauf braucht also nur noch bis zu dieser Array-Position zu gehen. Zur Beschreibung von Bubble-Sort siehe Aufgabe 30.

Aufgabe 47. Schreiben Sie eine Funktion `mvmultiplication`, die die Matrix-Vektor-Multiplikation einer spaltenweise gespeicherten Matrix $A \in \mathbb{R}^{m \times n}$ mit einem Vektor $x \in \mathbb{R}^n$ realisiert und den Ergebnisvektor $b = Ax \in \mathbb{R}^m$ zurückgibt. Die Dimensionen $m, n \in \mathbb{N}$ können Konstanten im Hauptprogramm sein, müssen aber als Parameter an die Funktion übergeben werden. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem A und x eingelesen werden und $b = Ax$ ausgegeben wird.

Aufgabe 48. Gegeben seien Polynome $p(x) = \sum_{j=0}^m a_j x^j$ und $q(x) = \sum_{j=0}^n b_j x^j$ in Form ihrer Koeffizientenvektoren $a \in \mathbb{R}^{m+1}$ und $b \in \mathbb{R}^{n+1}$. Dann ist die Summe $r = p + q$ ein Polynom vom Grad $\max\{m, n\}$. Schreiben Sie eine Funktion, die für gegebene Koeffizientenvektoren a und b den Koeffizientenvektor c von r berechnet und zurückgibt. Insbesondere soll die Funktion also den dynamischen Vektor c anlegen. Im aufrufenden Hauptprogramm sollen $m, n \in \mathbb{N}_0$ sowie die dynamischen Vektoren $a \in \mathbb{R}^m$ und $b \in \mathbb{R}^n$ eingelesen werden und c ausgegeben werden.

Aufgabe 49. Man modifiziere das Bisektionsverfahren aus Aufgabe 33 so, dass nicht nur die approximative Nullstelle $a \approx z_0$, sondern die gesamte Folge (a_0, \dots, a_N) von Approximationen zurückgegeben wird, d.h. man speichere in jedem Bisektionsschritt die linke Intervallgrenze.

Aufgabe 50. Schreiben Sie eine Funktion `primfaktoren`, die für eine natürliche Zahl $n \in \mathbb{N}$ deren Primfaktoren bestimmt und als Vektor $p \in \mathbb{N}^k$ zurückgibt. Die Koeffizienten p_j des Vektors $p \in \mathbb{N}^k$ sind also Primzahlen, und es gilt $n = \prod_{j=1}^k p_k$. Um eine Liste aller möglichen Primfaktoren zu erhalten, verwende man das Sieb des Eratosthenes aus Aufgabe 44.