

3. Übungsblatt zur Einführung in das Programmieren mit Matlab/GNU Octave

- (1) Schreiben Sie eine Funktion `qua`, die als Eingabe-Argument eine Zahl q entgegen nimmt und deren Quadrat auf dem Bildschirm ausgibt, und speichern Sie die Funktion unter dem Namen `qua.m` ab. Testen Sie die Funktion mindestens mit den Werten 9, -10, 0, 'aha' vom Command Window aus.
- (2) Ändern Sie die Funktion so, dass sie nichts auf dem Bildschirm ausgibt, sondern stattdessen das Quadrat von q als Ausgabe-Argument zurück gibt. Rufen Sie die Funktion im Command Windows folgendermaßen auf: $r = \text{qua}(t)$, wobei t für irgendeine Zahl steht, z.B. eine der oben erwähnten. Zeigen Sie r an.
- (3) Schreiben Sie ein Skript, welches die Funktion `qua()` mehrmals mit verschiedenen Argumenten aufruft. Entnehmen sie die Werte für die Argumente einem Vektor.
- (4) Erweitern Sie das die Funktion aufrufende Skript so, dass die von `qua()` zurückgegebenen Werte aufsummiert werden. Ziehen Sie aus der Summe am Ende die Wurzel. Was macht ihr Skript eigentlich?
- (5) Ändern Sie das Programm aus Aufgabe 10 des 1. Übungsblattes so ab, dass es zu einer Funktion wird, die als Eingabe-Parameter q und n entgegen nimmt, wobei q die zu entwickelnde Zahl ist und n angibt, wie weit entwickelt werden soll. Zurückliefern soll die Funktion das Ergebnis der Berechnung, wie es schon bisher in der ursprünglichen Aufgabe berechnet wurde. Die Funktion selbst soll nach den abschließenden Tests keine Ausgaben mehr auf dem Bildschirm erzeugen.
- (6) Erstellen Sie wiederum ein Skript, welches diese Funktion mehrfach aufruft, und die Ergebnisse in einem Vektor v speichert, so dass in $v(i)$ das Resultat des i -ten Aufrufs der Funktion steht. Geben Sie zum Schluss diesen Vektor aus.
- (7) Erweitern sie das Skript so, dass der Ergebnis-Vektor in einer Datei abgespeichert wird.
- (8) Löschen Sie mit `clear` alle Daten im *Command Window*. Laden Sie nun im Command Window die abgespeicherte Datei und geben sie den Inhalt des Vektors aus.
- (9) Kopieren Sie die Funktion in eine Datei anderen Namens (ändern Sie entsprechend den Namen der Funktion). Ändern Sie diese neue Funktion so ab, dass die Funktion immer die erste und die letzte Partialsumme S_1 und S_n zurück gibt in Form zweier Skalare. Rufen Sie diese Funktion für mehrere q auf und speichern Sie auch hier die beiden Rückgabewerte in zwei Vektoren z.B. `q1` und `qn`. Sorgen Sie dafür, dass nach dem Testen nur noch die Ausgabe der beiden Vektoren sowie ein diesen beiden Ausgaben vorangestellter Kommentar auf dem Bildschirm ausgegeben werden.
- (10) Wandeln Sie Ihre Skripte zur Multiplikation von Matrizen mit Matrizen und Matrizen mit Vektoren in Funktionen um.
- (11) Schreiben Sie ein Skript, welches eine Matrix A definiert, eine Matrix B sowie einen Vektor b , so dass die Operation ABb möglich ist. Rufen Sie nun im Skript die entsprechenden Funktionen auf, so dass ABb berechnet wird.
- (12) Überlegen Sie welche Kommentare und welche Ausgaben sinnvoll sind, um Ihren Kollegen, die nicht an diesen Kurs teilnehmen und etwa so viel Matlab beherrschen wie Sie, schnell eine Übersicht über diese Übungsaufgabe bzw. speziell über Ihre Lösung zu vermitteln und sie davon zu überzeugen, dass ihre Lösung richtig ist und es sich nicht lohnt nach Fehlern zu suchen.