

Matlab - Kompaktkurs

ANWESENHEITSAUFGABEN 2

Aufgabe 1 (Wir schreiben unser erstes M-File)

Schreiben Sie ein M-File `Kreis.m` welches den Radius `r` eines Kreises einliest, den Kreisumfang und den Flächeninhalt des Kreises berechnet und beides anschließend ausgibt. Gehen Sie dabei folgendermaßen vor:

- 1) Schreiben Sie am Anfang des M-Files eine Erläuterung zum Programm auf.
- 2) Lesen Sie den Radius des Kreises mit dem `input`-Befehl ein.
- 3) Geben Sie die eingegebene Zahl mittels `disp` und `num2str` wieder aus.
- 4) Berechnen Sie den Kreisumfang und den Flächeninhalt des Kreises. Vergessen Sie nicht Kommentare dazuzuschreiben was Sie gerade im Programm berechnen.
- 5) Nutzen Sie die Befehle `disp` und `num2str` um den Kreisumfang und den Flächeninhalt des Kreises auszugeben.

Aufgabe 2 (Funktionen 1)

Schreiben Sie Funktionen `FlaecheKreis(r)` und `UmfangKreis`, und ersetzen Sie die Berechnung von Fläche und Umfang des Kreises in ihrem M-File `Kreis.m` durch Aufrufe dieser Funktionen!

Aufgabe 3 (Funktionen 2)

Schreiben Sie ein Programm welches mittels eines Funktionsaufrufs die Summe und den Mittelwert eines Vektors bestimmt. Folgen Sie dabei folgender Anleitung:

- 1) Schreiben Sie eine Funktion `MittelwertSumme.m` welche für einen Vektor `v` dessen Summe und dessen Mittelwert berechnet. Überlegen Sie zunächst was die Argumente und was die Rückgabewerte der Funktion sind. Berechnen Sie dann die Summe und den Mittelwert von `v` mit geeigneten Vektorfunktionen.
- 2) Schreiben Sie ein Hauptprogramm `FunktionVektor.m`. Gehen Sie dabei wie folgt vor:
 - a) Schreiben Sie am Anfang des Hauptprogramms einen Kommentar zum Programm.
 - b) Lesen Sie den Vektor mit dem `input`-Befehl ein.
 - c) Rufen Sie die Funktion `MittelwertSumme` mit dem richtigen Rückgabewert und den richtigen Argumenten auf (welche sind das?).
 - d) Geben Sie die Summe und den Mittelwert aus.

Aufgabe 4 (if-Anweisung)

Schreiben Sie ein M-File welches das Vorzeichen einer gegebenen Zahl bestimmt:

$$\text{sign}(x) = \begin{cases} +1 & \text{if } x > 0 \\ -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \end{cases}$$

Gehen Sie wie folgt vor:

- 1) Schreiben Sie am Anfang des M-Files eine Erläuterung zum Programm auf.
- 2) Lesen Sie eine Zahl ein.
- 3) Schreiben Sie eine `if`-Anweisung, die das Vorzeichen der eingelesenen Zahl bestimmt und die Variable `Signum` mit diesem Vorzeichen belegt.
- 4) Führen Sie Ihr Programm aus und prüfen ob es für eingegebene reelle Zahlen das richtige Ergebnis liefert.

Aufgabe 5 (switch-Anweisung)

Schreiben Sie ihr Programm aus Aufgabe 4 um indem Sie nun eine `switch` statt einer `if` Anweisung benutzen (Achtung: Das ist etwas schwieriger und nicht unbedingt ein gutes Beispiel für die Verwendung der `switch` Anweisung!).

Aufgabe 6 (for-Schleifen)

Schreiben Sie ein Programm welches mittels einer `for`-Schleife die Fakultät einer gegebenen natürlichen Zahl bestimmt. Folgen Sie dabei folgender Anleitung:

- 1) Schreiben Sie am Anfang des M-Files eine Erläuterung zum Programm auf.
- 2) Weisen Sie einer Variablen `n` eine natürliche Zahl zu.
- 3) Schreiben Sie eine `for`-Schleife mit der Sie die Fakultät der gegebenen Zahl `n` berechnen können.
- 4) Geben Sie die Fakultät der Zahl aus.

Aufgabe 7 (while-Schleifen)

Zur Approximation der Kreiszahl π könnte man folgende Reihe benutzen:

$$\frac{\pi^2 - 8}{16} = \sum_{n=1}^{\infty} \frac{1}{(2n-1)^2(2n+1)^2}. \quad (1)$$

Bricht man die Reihe nach dem k . Folgenglied ab, so erhält man die k -te Partialsumme

$$S_k = \sum_{n=1}^k \frac{1}{(2n-1)^2(2n+1)^2}.$$

Schreiben Sie ein Programm welches mittels einer `while`-Schleife und obiger Formel (1) eine Näherung für π mit einem Fehler von höchstens $tol = 10^{-12}$ bestimmt. Gehen Sie dabei wie folgt vor:

- 1) Schreiben Sie am Anfang des M-Files eine Erläuterung zum Programm auf.

- 2) Initialisieren Sie die Variablen, welche Sie in der `while`-Schleife benötigen.
- 3) Schreiben Sie eine `while`-Schleife, welche bis zur gegebenen Toleranz $tol = 10^{-12}$ die aktuelle Iteration, die aktuelle Partialsumme, den aktuellen Näherungswert `piApprox` für π und den aktuellen Approximationsfehler $|\pi - \text{piApprox}|$ bestimmt.
- 4) Geben Sie die Anzahl der benötigten Iterationen `N` und den Approximationsfehler nach der `N`. Iteration aus.