

---

# Einführung in die Informatik I

## Übung 3

### 1 Einführung in die Vektoren

Welche der folgenden Ausdrücke sind ohne Fehlermeldung gültig? Geben Sie gegebenenfalls das Ergebnis an. Es sei  $V = [1.1, 2.2, 3.3]$  und  $W = [3, 2, 1]$

`W(floor(V(3)))`       `V(W(1))`       `V(W(V(1)))`       `W(abs(V(3)))`

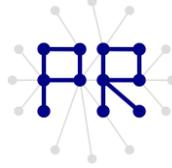
### 2 Diagonalmatrix: Kennen Sie die `linspace`- und `diag`-Befehle?

- Erzeugen Sie einen Vektor  $v$  und füllen Sie ihn mit  $n$  1en.
- Erzeugen Sie eine (quadratische) Diagonalmatrix  $A$  mit dem Vektor  $v$  auf der Diagonalen.
- Kopieren Sie die Diagonale von  $A$  in einen Vektor  $w$ .

### 3 Spiegeln eines Arrays

Mit Hilfe einer `for`- oder `while`-Schleife soll ein vorgegebenes (eindimensionales) Array umgedreht/gespiegelt werden. Dabei darf kein zweites Array verwendet werden – d.h. das Spiegeln des Arrays darf nur durch Vertauschungen innerhalb des Arrays erfolgen!

**TIPP:** Denken Sie zunächst darüber nach, wie man den Wert zweier Variablen durch Wertzuweisung miteinander vertauschen kann (es wird eine **Hilfsvariable** und **Kreisbewegung** der Variablen benötigt).



## 4 Palindrome

Mit Hilfe einer `for`- oder `while`-Schleife und einer `if`-Anweisung lässt sich testen, ob die Komponenten eines Arrays ein sogenanntes Palindrom bilden. Ein Palindrom ist ein Wort (oder Satz), das sich vorwärts und rückwärts gelesen gleich, oder die gleiche Bedeutung besitzt. Beispiele sind die Wörter: ANNA und RENTNER oder auch der Satz: Nie grub Ramses Marburg ein. Das Array bzw. der Vektor  $v = [1 \ 2 \ 3 \ 4 \ 5 \ 4 \ 3 \ 2 \ 1]$  kann auch als solches Palindrom angesehen werden und im Gegenteil das Array  $[1 \ 0 \ 0 \ 4 \ 3 \ 2 \ 1]$  ist kein Palindrom.

a) Überlegen Sie, wie man mit Hilfe einer Schleife einen einfachen Palindrom-Test programmieren kann.

b) Schreiben Sie eine MATLAB-Funktion namens `ist_palindrom`, der ein Array übergeben wird. Wenn es sich bei dem Array um ein Palindrom handelt, soll die Funktion die Zahl 1 zurückgeben, ansonsten 0. Die Schleife soll mit einer Fehlermeldung abgebrochen werden, sobald klar ist, dass es sich bei dem Array nicht um ein Palindrom handelt.

**Achtung:** Die Anzahl der Elemente eines Arrays kann eine gerade oder auch eine ungerade Zahl sein. Berücksichtigen Sie dies bei der Entwicklung der Lösung.

Sie können Ihre Lösung dann überprüfen indem Sie die Funktion aufrufen mit z. B.:

```
>> A = [1 2 3 4 4 3 2 1]
>> y = ist_palindrom(A)
```



## 5 Zusatzaufgabe

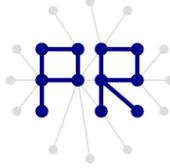
Kreuzen Sie die jeweils richtigen Aussagen an

### a) if-Verzweigungen

- if-Abfragen müssen durch ein 'end' geschlossen werden.
- Die Bedingung der if-Abfrage befindet sich in der zweiten Zeile.
- Für jede "elseif"-Verzweigung wird ein weiteres 'end' benötigt.
- if-Abfragen müssen nicht immer eine Anweisungssequenz ausführen (d.h. können übersprungen werden).

### b) "for-while"-Schleifen

- for Schleifenvariable=Start:Schritt:Ende  
    Schleifensequenz  
    end
- for Schleifenvariable=Start:Ende:Schritt  
    Schleifensequenz  
    end
- Innerhalb einer "for"-Schleife muss die Variable beeinflusst werden.
- Innerhalb einer "while"-Schleife muss die Variable beeinflusst werden.
- Eine "for"-Schleife kann zur Endlosschleife werden.
- Eine "while"-Schleife kann zur Endlosschleife werden.
- "for"- und "while"-Schleifen müssen durch 'end' geschlossen werden.
- In einer "while"-Schleifenbedingung liegt eine logische Abfrage vor.



c) Skripte vs. Funktionen

- Funktionen müssen mit `'function'` begonnen werden.
- Skripte müssen mit `'script'` begonnen werden.
- Eine Funktion muss mit `'end'` geschlossen werden.
- Funktionsname und Dateiname müssen übereinstimmen.
- Funktionen werden über Funktionsnamen inklusive Eingabeparameter aufgerufen.
- Skripte werden über den Dateinamen aufgerufen.