

**DHBW Stuttgart**

Fakultät Technik, Maschinenbau

# Programmmentwurf in C

Kurs:

Modul:

Dozent:

Datum:

Bearbeitungszeit: 180 Minuten

Hilfsmittel: C-Referenzkarte

**Allgemeine Hinweise:** Die Prüfungsleistung beinhaltet den Entwurf von 2 Programmen (Aufgabe 1 und 2). Beide Aufgaben sind in der vorgegebenen Zeit zu bearbeiten. Die Aufgaben sind in Teilaufgaben mit steigendem Schwierigkeitsgrad aufgeteilt. Sie können beide Aufgaben unabhängig voneinander bearbeiten, die Reihenfolge der Teilaufgaben ist jedoch einzuhalten. Jede Aufgabe ist in einer separaten Quelldatei mit dem Namen Aufgabe1\_MATNR.c bzw. Aufgabe2\_MATNR.c abzulegen. Ersetzen Sie dabei das Kürzel MATNR durch Ihre Matrikelnummer. Als Entwicklungsumgebung ist Dev-C++ zu verwenden.

## Aufgabe 1

Implementieren Sie jede der folgenden Teilaufgaben in einer separaten Funktion. Sofern nicht anders beschrieben, soll die Funktion den errechneten Wert zurückliefern. Nachdem Sie eine Teilaufgabe bearbeitet haben, fügen Sie in der `main()`-Funktionen einen exemplarischen Aufruf der entsprechenden Funktion ein, sowie eine Ausgabe des errechneten Ergebnisses auf der Konsole.

In dieser Aufgabe ist ein C-Programm zur Auswertung einer Reihe von Messwerten zu erstellen.

- a) *Eingabe*: Alle Messwerte sollen dabei als Typ `double` vom Benutzer eingegeben und in einem Array abgelegt werden. Die Anzahl der zu verarbeitenden Werte kann in jedem Programmablauf unterschiedlich sein und ist im Voraus nicht bekannt.
- b) *Ausgabe*: Alle eingegebenen Messwerte sollen in angemessener Formatierung ausgegeben werden.
- c) *MinMax*: Entwickeln Sie eine Funktion zur Bestimmung von Minimum und Maximum der Messreihe.
- d) *Mittelwert*: Berechnen Sie den Mittelwert der Messreihe. Zunächst ist eine Funktion zur Berechnung der Summe aller Messwerte erforderlich. Schreiben Sie ebenfalls entsprechende Funktion *Summe* hierfür.  
Der Mittelwert  $mw$  über  $n$  Elemente errechnet sich wie folgt:

$$mw = \frac{1}{n} \sum_{i=1}^n x[i]$$

- e) *Varianz*: Berechnen Sie Streuung bzw. Varianz der Messreihe. Die Varianz  $var$  über  $n$  Elemente errechnet sich wie folgt:

$$var = \frac{1}{n-1} \sum (x[i] - mw)^2 = \frac{1}{n-1} \left( \sum_{i=1}^n (x[i])^2 - n * mw * mw \right)$$

- f) *Sortieren*: Implementieren Sie eine Funktion zur aufsteigenden Sortierung aller Messwerte. Die Wahl des Sortieralgorithmus bleibt Ihnen überlassen.

## Aufgabe 2

Implementieren Sie jede der folgenden Teilaufgaben in einer separaten Funktion. Nachdem Sie eine Teilaufgabe bearbeitet haben, fügen Sie in der `main()`-Funktionen einen exemplarischen Aufruf der entsprechenden Funktion ein.

In dieser Aufgabe ist ein C-Programm zu erstellen, welches die Bauteile einer Reihenschaltung verwaltet. Als Datenstruktur soll eine verkettete Liste von Bauteil-Strukturen verwendet werden.

- a) Definieren Sie eine Struktur `Bauteil`. Diese soll als „Nutzlast“ (neben der Zeigervariablen `*next`) eine Bezeichnung und den Widerstand beinhalten, sowie eine weitere Eigenschaft Ihrer Wahl. Finden Sie geeignete Datentypen für die einzelnen Komponenten der Struktur. Der Strukturtyp ist global zu definieren, ebenso eine Zeigervariable `*start`, welche die Startadresse der verketteten Liste aufnimmt. Diese Adresse ist mit `NULL` zu initialisieren.

- b) *Eingabe*: Erstellen Sie eine Funktion mit dem Prototyp

```
void eingabe();
```

welche die Daten für ein Bauteil vom Benutzer einliest und das Bauteil als vorderstes Element in die Liste einfügt.

- c) *Ausgabe*: Erstellen Sie eine Funktion mit dem Prototyp

```
void ausgabe(Bauteil *p);
```

die alle Eigenschaften des Bauteils `*p` auf der Konsole ausgibt.

- d) *Schaltung*: Erstellen Sie eine Funktion mit dem Prototyp

```
void schaltung();
```

welche die in der verketteten Liste gespeicherten Bauteile auf dem Bildschirm ausgibt (benutzen Sie die Funktion aus Teilaufgabe c hierfür), sowie den Gesamtwiderstand über alle Bauteile der Liste berechnet und ausgibt.