

7 Funktionen - Übungen

StaticVariable (118)

- Was gibt dieses Programm aus?

```
int i;
for (i = 0; i < 5; i++) {
    int num = 5;
    num--;
    printf("%d\n", num);
}
```

- Ändern Sie das obige Programm durch Hinzufügen von *static* so ab, dass eine Ausgabe wie folgt ausgegeben wird:

```
4
3
2
1
0
```

GlobalVariable

- main.c:

```
#include "inc.h"
int globVar;           // Definition 1
int main()
{
    inc();
    printf("\nmain:%d", --globVar);
    return 0;
}
```

- inc.c:

```
int globVar;           // Definition 2
void inc() {
    printf("\ninc: %d", ++globVar);
}
```

- Hier wird zweimal *globVar* definiert. Wird der Code ausgeführt, dann sieht man allerdings, dass die gleiche Variable verwendet wird. Der Code wird auf diese Art sehr schlecht lesbar.

Sollen Variable wirklich global verwendet werden, dann ist es besser sich zu überlegen wo genau die globale Variable definiert werden soll und an allen anderen Stellen die Variable als `extern` zu definieren:

```
extern int globVar;     // Bedeutet: globVar ist woanders als int definiert
void inc() {
    printf("\ninc: %d", ++globVar);
}
```

Der Vorteil ist, dass damit geprüft wird ob es diese externe Variable auch wirklich gibt. Ohne `extern` könnte in `main` der Name der Variable unbeabsichtigt geändert werden und das Programm funktioniert immer noch.

A01_Add

- Addition von zwei double Zahlen `add(a, b)`
- Erstelle die Funktion in einer eigenen Datei `add.c`, binde die Datei über die Header-Datei `add.h` ein

L:

`main.c`:

```
#include <stdio.h>
#include <stdlib.h>
#include "add.h"

int main()
{
    printf("%0.2f + %0.2f = %0.2f\n", 3.1, 4.5, add(3.1, 4.5));
    return 0;
}
```

`add.h`:

```
#ifndef ADD_H_INCLUDED
#define ADD_H_INCLUDED
double add(double, double);
#endif // ADD_H_INCLUDED
```

`add.c`:

```
#include "add.h"
double add(double a, double b) {
    return a + b;
}
```

A02_Increment

- Inkrement einer Zahl - Call-By-Reference
- Aufruf mit `inc(&num)`
- Funktion in eigene Datei `inc.c`, Einbinden über die Header-Datei `inc.h`

A03_Fakultaet

- Die Fakultät einer Zahl spielt in der Wahrscheinlichkeitsrechnung aber auch in Taylorreihen eine Rolle. Berechne die Fakultät einer ganzen Zahl:
$$n! = 1 * 2 * 3 * \dots * n$$
- Erstelle die Funktion `int fact(int)` in einer eigenen Datei `fact.c` und binde sie über eine Header-Datei ein.

A04_Power

- Berechne die Hochzahl einer Zahl `double power(double num, int pow)`
- Berücksichtige Exponent 0.

A05_TempKonv

- Berechne die Temperatur in Fahrenheit aus der Temperatur in °C und umgekehrt.
 - $TF = TC * 1,8 + 32$ (`double degCtoDegF(double degC)`)
 - $TC = (TF - 32) * 5/9$ (`double degFtoDegC(double degF)`)

A06_Schaltjahr

- Erstelle eine Funktionen zur Berechnung ob ein eingegebenes Jahr ein Schaltjahr ist `boolean isSchaltjahr(int jahr)`
 - Ein Jahr ist ein Schaltjahr, wenn es restlos durch 4 teilbar ist (2000 ok)

- Falls sich die Jahrzahl durch 100 restlos teilen lässt, ist es kein Schaltjahr (2000 nok)
- Falls auch eine Teilung durch 400 ganzzahlig möglich ist, dann ist es ein Schaltjahr (2000 ok)

Die Regeln sind in dieser Reihenfolge anzuwenden. Daher ist 2000 ein Schaltjahr.

A07_Kreisfläche

- Berechne die Kreisfläche in einer Funktion ohne einen Rückgabewert zu verwenden (globale Variable).
- Erstelle die Funktion **void kreisFlaeche(double)** in einer eigenen Datei kreisFlaeche.c und binde sie über eine Header-Datei ein.

A08_Zufallsfeld

- Erstelle 3 Funktionen um ein Feld mit Zufallszahlen zwischen 0 und 100 (max. Größe 100) zu füllen:
 - mit statischer Variable **int* zufallsfeld(int groesse)**
 - mit mitgegebenem Feld **int* zufallsfeld(int *feld, int size)**
 - mit reservierten Speicher **int* zufallsfeld(int size)**

A09_MemLeak

- Erstelle eine Malloc-Funktion und "vergesse" den Speicher freizugeben. Setze die Funktion in eine Schleife und beobachte was passiert.
- Im Taskmanager (oder direkt in VS) kann der Speicherbedarf des Programms ermittelt werden.

A10_SortString

- sortiere die Zeichen einer Zeichenkette nach Alphabeth, wobei Großbuchstaben eines Buchstaben gleiche Priorität wie Kleinbuchstaben haben. Zum Beispiel gilt für folgende Zeichenkette "aBAcb" sortiert --> "aABbc"