

Funktionen (Call-by-Value), statische Variablen

4. Tutorium INF1 // Unterlagen: [http://floba.info/Dateien/Tutorium Informatik I](http://floba.info/Dateien/Tutorium%20Informatik%20I) // Florian Bauer
(florian.bauer@fh-bielefeld.de) // Versionsstand: 24.10.2010, 19:56

THEORIE

Notizen während des Tutoriums ergänzen...

Funktion OHNE Argumente und OHNE Rückgabewert

```
void myFunction(void); //Prototyp

int main()
{
    myFunction(); //nicht myFunction(void)!

    return 0;
}

void myFunction(void)
{
    if(1 > 2)
        return;

    int a = 3;
    int b = 4 * a;
    return; //optional
}
```

Funktion OHNE Argumente und MIT Rückgabewert

```
int myFunction(void);

int main()
{
    myFunction();
    int a = myFunction();

    return 0;
}

int myFunction(void)
{
    if(1 > 2)
        return 0;

    int a = 3;
    int b = 4 * a;
    return b; //irgendwo muss ein return <int> erreichbar sein!
}
```

Funktion MIT 1 Argument und OHNE Rückgabewert

```
void myFunction(int arg);

int main()
{
    int a = 1;
    myFunction(a);
    int b = a;

    myFunction(2);

    int c = myFunction(3); //Fehler, da kein Rückgabewert

    return 0;
}

void myFunction(int arg)
{
    if(arg > 2)
        return;

    int a = arg;
    int b = 4 * a;
    return;
}
```

Funktion MIT mehreren Argumenten und OHNE Rückgabewert

```
void myFunction(int arg1, int arg2, int arg3);

int main()
{
    int a = 1;
    myFunction(a,2,3);
    myFunction(4,5,6);

    int b = myFunction(a,a,a); //Fehler, da kein Rückgabewert

    return 0;
}

void myFunction(int arg1, int arg2, int arg3)
{
    if(arg1 > arg2)
        return;

    int a = arg3;
    int b = 4 * a;
    return;
}
```

Funktion MIT mehreren Argumenten und MIT Rückgabewert

```
int myFunction(int arg1, int arg2, int arg3);

int main()
{
    int a = 1;
    myFunction(a,2,3);
    int b = myFunction(a,2,3);

    return 0;
}

int myFunction(int arg1, int arg2, int arg3)
{
    if(arg1 > arg2)
        return arg3;

    int a = arg3;
    int b = 4 * a;
    return b;
}
```

Funktion mit statischer Variable

```
#include <stdio.h>
void myFunction(void);

int main()
{
    for(int i = 0; i < 10; i++)
        myFunction();

    return 0;
}

void myFunction(void)
{
    static int i = 0;
    i++;
    printf("Aufruf Nr. %d\n", i);
}
```

AUFGABEN

Aufgabe 1: Energie von Windkraftanlagen (WKA)

Die Energie einer WKA in MWh berechnet sich aus dem Produkt aus Leistung in kW, dem Umrechnungsfaktor von kW in MW und der Volllaststundenzahl. Schreibe eine Funktion die dies berechnet und zurückgibt.

- Speicher die Energiemenge in einer Variablen in der `main`-Funktion und gib den Wert mit `printf()` aus!
- Gib die Energiemenge in der `main`-Funktion mit `printf()` aus, ohne diese vorher in einer Variablen zu speichern!

Aufgabe 2: Primzahlen

Bereits im 2. und 3. Tutorium gab es eine Primzahlaufgabe (*Tutorium 2, Aufgabe 1* und *Tutorium 3, Aufgabe 4 h*). Der Quelltext soll nun wie folgt verändert in eine Funktion geschrieben und nach und nach erweitert werden¹:

- Innerhalb der Funktion sollen die Schaltjahre bestimmt und ausgegeben werden. Die Funktion soll weder Argumente übernehmen noch einen Rückgabewert liefern.
- Nun soll die Funktion die Anzahl der gefundenen Schaltjahre zurückgeben. Dieser Wert soll im Hauptprogramm mit `printf()` ausgegeben werden.
- Als Argumente soll die Funktion jetzt das Start- und das Endjahr übergeben bekommen. Zwischen den beiden Angaben soll die Funktion alle Schaltjahre ausgeben.

¹ Zur Lösung kann auch der Quelltext aus der Musterlösung verwendet werden.