

Zählschleifen mit while:

Die ‚while – Schleife‘ wird bei der strukturierten Programmierung in C relativ häufig verwendet. So lässt sich damit eine fest vorgegebene Anzahl von Schleifendurchläufen realisieren. Die Konstruktion einer Schleife mit ‚while‘ erinnert an das ‚if-Anweisung‘.

Wird **nur ein Befehl** abgearbeitet sieht das wie folgt aus:

while (*Ausdruck*)

Befehl;

Falls **mehrere Befehle** benötigt werden, werden diese einfach zwischen Klammern geschrieben:

while (*Ausdruck*)

{

Befehl 1;

...

Befehl n;

}

Befehle innerhalb der ‚while – Schleife‘



12. Übung: Probieren Sie die Funktion der while – Schleife des nachfolgenden C - Programms aus. Und beschreiben Sie die Funktion der while - Schleife in eigenen Worten. Welche Ausgabe erzeugt das Programm?

```
#include <stdio.h>

int main()
{
    int i;

    i = 10;
    while (i > 0) {
        printf("%d\n", i);
        i = i - 1;
    }
    getchar();
    return 0;
}
```

Lösung der Übung 12:

13. Übung: Analysieren Sie das nachfolgende Programm.
Welche Ausgabe wird erzeugt?

```
#include <stdio.h>

int main()
{
    int i;
    i = 10;
    while (i > 0) {
        printf("%d\n", i);
    }
    getchar();
    return 0;
}
```

Lösung der Übung 13:

Schleifenkontrolle mit **break** und **continue**:

Bei manchen Programmen ist es nützlich, aus einer Schleife unabhängig von der Schleifenbedingung ausbrechen zu können. Dies kann mit dem ‚break – Befehl‘ erfolgen.

14. Übung: Analysieren Sie das nachfolgende Programm.

Welche Funktion hat der ‚**break** – Befehl‘?

```
#include <stdio.h>
int main()
{
    int i;
    i = 10;
    while (1) {
        if (i < 0)
            break;
        printf("%d\n", i);
        i = i - 1;
    }
    getchar();
    return 0;
}
```

Lösung der Übung 14:

15. Übung: Analysieren Sie das nachfolgende Programm.

Welche Funktion hat der ‚**continue** – Befehl‘?

```
#include <stdio.h>

int main()
{
    int i;

    i = 100;
    while (i >= 0) {
        if (i % 10 != 0) {
            i = i - 1;
            continue;
        }
        printf("%d\n", i);
        i = i - 1;
    }
    getchar();
    return 0;
}
```

Lösung der Übung 15:

Frage: Welche Folgen hätte es, wenn wir **$i = i - 1$** vergessen würden?
(Versuchen Sie es bitte mal.)

16. Übung: Verwenden Sie in Übung 15 nun den **Dekrement** - Befehl.

Lösung der Übung 16: