

## 4.2 Von C nach C++/Default-Parameter

### 4.2.1 Eine Funktion mit unterschiedlich vielen Parametern

Im folgenden Programm wird die Funktion `berechne_laenge()` zweimal aufgerufen.

```

1 #include <iostream>
2 #include <math.h> // mit -lm uebersetzen!
3 using namespace std;
4
5 double berechne_laenge(double x, double y=0.0, double z=0.0)
6 {
7     return sqrt(x*x+y*y+z*z);
8 }
9 //=====
10 int main()
11 {
12     // Zaun 1: ebenerdig
13     cout << berechne_laenge(8.0, 12.0) << endl;
14
15     // Zaun 2: Hanglage
16     cout << berechne_laenge(8.0, 12.0, 5.0) << endl;
17
18     return 0;
19 }
```

Es fällt auf, dass die Funktion einmal mit zwei Parametern und einmal mit drei Parametern aufgerufen wird. Das gibt es in C++-Programmen immer wieder. Drei Ursachen sind möglich:

- Schon in ANSI-C89 gibt es die Möglichkeit, für Funktionen wie `printf()` Parameterlisten mit variabler Länge einzusetzen. Dazu dienen die in `<stdarg.h>` definierten Makros. Das Verfahren ist aber umständlich und wird nicht oft genutzt.<sup>1</sup>
- Das Überladen von Funktionen (hier nicht benutzt)
- So genannte Default-Parameter

### 4.2.2 Default-Parameter

Die Funktion `berechne_laenge()` hat *Default-Parameter* (=Standard-Parameter).

Default-Parameter einer Funktion werden beim Erstellen der Funktion mit einem Standard-Wert (hier: `0.0`) vorbelegt. Sie können beim Aufruf der Funktion weggelassen werden. Sie eignen sich daher besonders für solche Parameter, die in der Regel einen ganz bestimmten Wert haben. Man könnte sie mit Optionen auf der Kommandozeile vergleichen, die auch immer nur dann angegeben werden müssen, wenn man etwas Besonderes braucht.

In C++ wird ein Default-Parameter wie die Initialisierung einer Variablen geschrieben:

```

1 void funktion(int a=3, char *s="", void *p=NULL);
```

Dabei können immer nur der letzte, die letzten zwei oder die letzten  $N$  Parameter vorbelegt werden.

Beim Aufruf können dann immer nur der letzte, die letzten zwei oder die letzten  $N$  Parameter weggelassen werden.

<sup>1</sup>Eine ältere Methode nutzt dafür Makros aus der Datei `C<stdarg.h>`.

### 4.2.3 Wohin gehören Default-Parameter?

Kann oder muss man die Default-Werte in die Funktionsdefinition oder in den Prototyp schreiben oder beides?

- Die Default-Werte dürfen nur an **einer** Stelle eingefügt werden.
- Bevor man die Funktion mit weniger Parametern als maximal möglich benutzt, müssen die Default-Werte bekannt sein. Das liegt an der bekannten Tatsache, dass C und C++ Ein-Pass-Compiler ermöglichen, so dass jede Funktion (und jeder Parameter) bei ihrer Benutzung bereits bekannt sein müssen.

Daher bietet sich — wenn man schon Prototypen benutzt — als Grundregel an: **Default-Werte gehören in den Prototypen einer Funktion.**