Kapitel 5

Arithmetische Operatoren

Arithmetische Operatoren

- Man unterscheidet unäre und binäre Operatoren.
- Je nachdem, ob sie auf einen Operanden wirken, oder eine Verknüpfung zweier Operanden bewirken.

Arithmetische Operatoren

- Wie die Berechnung durchgeführt wird, entscheiden die Operatoren.
- Sind bei der Division beide Operanden ganze Zahlen, ist es eine „Ganzzahl-Divison".
- Ist bei der Division hingegen mindestens ein Operand eine Gleitpunktzahl, so ist es eine „Gleitpunkt-Divison".

Arithmetische Operatoren

Operatoren	Bedeutung	Bemerkung
++	Inkrement	unär
--	Dekrement	unär
+	Vorzeichen Plus	unär
-	Vorzeichen Minus	unär
*	Multiplikation	binär
/	Division	binär
%	Modulo	binär, ganzzahlig
+	Addition	binär
-	Subtraktion	binär

Merkwürdige Ganzzahl-Arithmetik

- Es gibt einen Unterschied zwischen dem, was unser Taschenrechner und dem, was der Computer berechnet.

Operation	Taschenrechner	Computer
7/2	3.5	3
1.5 + 7/2	5	4.5
1/2 + 1/2		
1.0/2.0 + 1.0/2.0		
1/2 + 1.0/2.0		

Operatorenliste

Operatoren	Auswertrichtung	Priorität	Bemerkung
() [] -> .	li -> re	hoch	Klammer, Indexop, Komponentenop
! ~ ++ -- + - * & (Typ)	re -> li		unäre Operatoren
* / %	li -> re		arithmetische Operatoren
+ -	li -> re		arithmetische Operatoren
<< >>	li -> re		Shift-Operatoren
< <= > >=	li -> re		Vergleichsoperatoren
== !=	li -> re		Vergleichsoperatoren
&	li -> re		Bitoperatoren
^	li -> re		Bitoperatoren
|	li -> re		Bitoperatoren
&&	li -> re		logische Operatoren
||	li -> re		logische Operatoren
?:	li -> re		Auswahloperator
= += -= *= /= %= etc.	re -> li		Zuweisungsoperator
,	li -> re	niedrig	Ausdrucksfolge

Arithmetische Ausdrücke

- Ein Ausdruck kann im einfachsten Fall eine Konstante oder eine Variable sein.
- Meist handelt es sich aber um eine Berechnung unter Verwendung von Operatoren.

```
7.2
anzahl
a + b + c
```

Arithmetische Ausdrücke

- Vor dem Ausführen einer binären Operation werden die Datentypen der Operanden angeglichen.
- Die Anpassung erfolgt in der Weise, dass der „kleinere" Datentyp auf den „größeren" Datentyp erweitert wird.
- Das Ergebnis hat dann den „größeren" Datentyp.

char -> unsigned char -> short -> int -> unsigned int -> long -> unsigned long -> float -> double -> long double

Arithmetische Ausdrücke - Beispiele

Ausdruck	Ergebnis
7 + 1.2	8.2
2.0 * 8	16.0
1 / 2	0
9 / 2	4
1.0 / 3 * 6	2.0
2.5 + 3.0 * 2.0 – 0.5	8.0
2 * 3 % 4 + 1	3

Arithmetische Wertzuweisung

- Einer arithmetischen Variablen kann der Wert eines arithmetischen Ausdrucks zugewiesen werden.
- Der Zuweisungsoperator ist das = Zeichen.
- Hat die Variable auf der linken Seite des Zuweisungsoperators einen anderen Datentyp als der Ausdruck auf der rechten Seite, so wird die rechte Seite an den Datentyp der linken Seite angeglichen.

```
int i = 1.25                    // i = 1
float x = 7.1E+122              // Compiler-Warnung
```

Arithmetische Wertzuweisung

- Eine Wertzuweisung ist selbst wieder ein Ausdruck und liefert den zugewiesenen Wert.

```
z = 12
```

- ist ein Ausdruck und liefert den Wert **12**.

```
x = (y = (z = 12));
x = y = z = 12;
```

Arithmetische Wertzuweisung

Schreibweise	Bedeutung
x += 1	x = x + 1
x -= 1	x = x -1
x *= 2	x = x * 2
x /= 2	x = x / 2
x %= 2	x = x % 2
x++	Erhöht x nach der Verwendung um 1
++x	Erhöht x vor der Verwendung um 1
x--	Verringert x nach der Verwendung um 1
--x	Verringert x vor der Verwendung um 1

Mathematische Funktionen

- In der Standard-Header-Datei *cmath* sind mathematische Funktionen aufgeführt.

```
y = sqrt(x);
z = sin(w);
a = pow(2,10);
```

(Wobei Visual C++ bei Einbindung von <iostream> viele, viele weitere Bibliotheken automatisch auch hinzufügt.)

Mathematische Funktionen

Funktionsprototyp	Bedeutung
double sin(double x);	Sinus; x im Bogenmaß
double cos(double x);	Cosinus; x im Bogenmaß
double tan(double x);	Tangens; x im Bogenmaß
double atan(double x);	Arkus Tangens liefert Winkel im Bogenmaß
double exp(double x);	e^x
double log(double x);	ln(x) (für x > 0)
double pow(double x, double y);	x^y (für (x != 0 oder y > 0) und (x >= 0 oder y ganzzahlig))
double sqrt(double x);	Quadratwurzel von x (für x >= 0)
double fabs(double x);	|x| Absolutbetrag

Kommaoperator

- Mit dem Kommaoperator können dort mehrere Ausdrücke angegeben werden, wo sonst nur ein Ausdruck stehen darf.
- Die Ausdrücke werden von links nach rechts ausgewertet.
- Der Wert des Kommaoperator-Ausdrucks hat den Wert und den Typ des am weitesten rechts stehenden Einzelausdrucks.

Kommaoperator

```
#include <iostream>
using namespace std;
void main()
{
        int a, b, c;
        a = (b = 2, c = 3, b * c);
        cout << a << endl;
}
```

```
6
```