

# Konvertierungen

Den Umwandlungsregeln liegt die Zielsetzung zugrunde, bei der Umwandlung von Werten keine Information zu verlieren. Die Umwandlung erfolgt also immer zu einem Typ hin, der mehr Information aufnehmen kann.

Die Tabelle zeigt die Umwandlung der Datentypen. Je nach dem Typ der jeweiligen Operanden lässt sich ablesen, in welchen Datentyp die Operanden umgewandelt werden. Das Ergebnis des Ausdrucks hat den jeweils gleichen Typ.

Op1 Op2	long double	double	float	unsigned long long	unsigned int short char		
long	long						
double	double						
double		double					
float			float				
unsigned long				unsigned long			
long					long		
unsigned						unsigned	
int short char							int

Diese Konvertierung findet bei Ausdrücken statt, an denen nur arithmetische Datentypen beteiligt sind. Sind Operanden an einem Ausdruck beteiligt, die selbstdefinierte Datentypen haben, so gibt es dafür keine festgelegten Regeln.

Bei der Zuweisung von Werten an Datenobjekte kann es vorkommen, daß Datentypen so konvertiert werden, daß Information verloren geht. Dies ist immer dann der Fall, wenn Datenobjekte mit größerem Wertebereich umgewandelt werden in Datenobjekte mit kleinerem Wertebereich. Dabei gelten folgende Regeln:

Bei der Umwandlung von nichtganzzahligen Werten in ganzzahlige Werte wird der Bruchanteil abgeschnitten.

Die Umwandlung ganzzahliger Werte in Typen mit kleineren Wertebereich ist abhängig von der Implementierung, im Regelfall werden die höherwertigen Bits unterdrückt.

An einen Zeiger auf eine Basisklasse können Zeiger der Basisklasse oder davon abgeleiteter Klassen zugewiesen werden. Dabei findet implizit eine Konvertierung des Zeigertyps statt. Die explizite Konvertierung ist in jedem Fall möglich.

Erfolgt eine implizite Konvertierung in Verbindung mit einer Bedingung eines if-Statements oder einer Iteration, so ist der Ergebnistyp bool.

In Verbindung mit einem switch-Statement ist der Ergebnistyp ganzzahlig.