

INHALTSVERZEICHNIS

Vorwort	1
1 Einleitung	3
2 Ein einführendes Beispiel	6
2.1 Kommentare	8
3 Deklarationen/Definitionen	11
4 Typen, Konstanten, Operatoren, Ausdrücke	13
4.1 Typen	13
4.1.1 Elementare Typen und Typkonvertierung	13
4.1.1.1 Implizite Typkonvertierung	14
4.1.1.2 Explizite Typkonvertierung	16
4.1.2 Abgeleitete Typen	17
4.1.2.1 Referenz	19
4.1.2.2 Zeiger	19
4.1.2.3 Vektoren	20
4.1.2.4 Der spezielle Typ "void"	24
4.1.3 Typedef	26
4.2 Konstanten	27
4.2.1 Integer-Konstanten	27
4.2.2 Character-Konstanten	28
4.2.3 Reelle Konstanten	29
4.2.4 Strings	29

4.2.5	Const	30
4.2.6	Aufzählungen	31
4.3	Operatoren	33
4.3.1	Arithmetische Operatoren	34
4.3.2	Vergleichsoperatoren und boolesche Operatoren	35
4.3.3	Inkrement- und Dekrement-Operatoren	36
4.3.4	Bitweise logische Operatoren	37
4.3.5	Zuweisungsoperatoren	38
4.4	Ausdrücke	40
4.4.1	Bedingter Ausdruck	40
4.4.2	Kommaoperator	41
5	Anweisungen	43
5.1	Elementare Anweisungen, Blockstruktur, Gültigkeitsbereich von Variablen	43
5.2	Kontrollanweisungen	46
5.2.1	if-else	46
5.2.2	switch	48
5.2.3	while und for	49
5.2.4	do-while	51
5.2.5	break	51
5.2.6	continue	52
5.2.7	gotos und labels	53
6	Funktionen	54
6.1	Definition einer Funktion	54
6.2	Parameterübergabe	56
6.2.1	Strong Type Checking	57
6.2.2	call by value	58
6.2.3	call by reference	59
6.2.4	Vektoren als Parameter	60

6.3	Ergebnisrückgabe	62
6.3.1	Der Freispeicher	64
6.4	Weitere Parameterübergabemöglichkeiten	67
6.4.1	Default Argumente	67
6.4.2	Funktionen als Parameter	68
6.4.3	Ellipsen	70
6.5	Overloading von Funktionen	71
6.6	Inline-Funktionen	74
6.7	Die Funktion main und Hinweise zur Programmstruktur	75
6.7.1	Programmstruktur	76
7	Structures	80
8	Klassen	84
8.1	Motivation für das Klassenkonzept	84
8.2	Definition von Klassen und Member-Funktionen	90
8.2.1	Zeiger auf Klassenmember	92
8.2.2	Statische Klassenmember	94
8.2.3	Der this-Zeiger	97
8.2.4	Member-Funktionen	98
8.3	Gültigkeitsbereiche bei Verwendung von Klassen	101
8.4	Initialisierung von Klassen	104
8.4.1	Konstruktoren	104
8.4.2	Weitere Möglichkeiten zur Initialisierung	105
8.5	Löschen von Klassenobjekten	109
8.5.1	Destruktoren	109
8.6	Friends	112
8.7	Klassen als Member von Klassen	115
8.7.1	Konstruktoren/Destruktoren für Member-Klassen	115
8.8	Vektoren von Klassen	119
8.9	Structures und Unions	120

8.10	Bitfelder	122
9	Abgeleitete Klassen	123
9.1	Manipulation von Klassenobjekten	130
9.2	Klassenhierarchien	133
9.3	Zugriff auf vererbte Member	137
9.4	Konstruktoren/Destruktoren für abgeleitete Klassen	138
9.4.1	X(const X&) bei abgeleiteten Klassen	141
9.5	Virtuelle Funktionen	143
9.6	Virtuelle Destruktoren	149
9.7	Virtuelle Basisklassen	150
10	Operator Overloading	155
10.1	Möglichkeiten und Einschränkungen	156
10.1.1	Operator []	159
10.1.2	Operator ()	160
10.1.3	Operator =	161
10.1.4	Operator ->	162
10.1.5	Operatoren <i>new</i> und <i>delete</i>	163
10.2	Selbstdefinierte Typkonvertierung	164
11	Ein-/Ausgabe	169
11.1	Unformatierte Ein-/Ausgabe	169
11.2	Formatierte Ausgabe	174
11.3	Dateioperationen	178
12	Anhang	184
12.1	Tabelle der Operatoren	184
12.2	Tabelle der reservierten Worte	187
12.3	Tabelle der besonderen Character	187
12.4	Tabelle der Anweisungen	188

12.5	Tabelle der Ausdrücke	189
12.6	Hinweise zur Benutzung von UNIX-Rechnern	192
12.7	Hinweise zum Compiler	193
12.7.1	Aufruf des C++-Compilers	193
12.7.2	Compiler-Anweisungen	195
12.8	Unterschiede zur C++-Version 1.2	197
12.8.1	Änderungen der Semantik von Version 1.2	197
12.8.2	Nicht unterstützte Konzepte in Version 1.2	198
12.9	Zukünftige Neuerungen von C++	199
13	Aufgaben	201
14	Musterlösungen	209
15	Literatur	257
16	Register	266