

Inhaltsverzeichnis

1 Einleitung	9
1.1 Ist dieses Buch für Sie?	9
1.2 Aufbau	10
1.3 Compiler	11
1.4 Schreibweisen	12
1.5 Dank	12
1.6 Kontakt	13
1.7 Neue Features im Überblick	13
2 Sprache	19
2.1 Automatische Typableitung mit „auto“	19
2.2 Nachgestellter Ergebnistyp	23
2.3 Streng typisierte Aufzählungen	25
2.4 Ausnahmespezifikation	29
2.5 Attribute	31
2.6 „Rvalue“-Referenzen	34
2.7 Explizites Überschreiben	42
2.8 Ableitung und Überschreibung verhindern	45
2.9 Automatische Implementierung löschen und wiederherstellen	47
2.10 Konstruktoraufruf in Konstruktoren	52
2.11 Konstruktoren vererben	56
2.12 Unbeschränkte Union	58
2.13 Bereich-basierte „for“-Schleife	61
2.14 Initialisierer-Listen	65
<hr/>	
C++17	5

2.15 Einheitliche Initialisierung	68
2.16 Initialisierer für Elementvariablen	71
2.17 Lambda-Funktionen	72
2.18 Initialisierte Lambda-Bindungen	80
2.19 Generische Lambda-Funktionen	82
2.20 Konstante Ausdrücke	83
2.21 Erweiterte „friend“-Deklaration	89
2.22 Binäre Literale	91
2.23 Zahlentrennzeichen	91
2.24 Zeichenketten-Literale	92
2.25 Benutzerdefinierte Literale	96
2.26 „nullptr“	100
2.27 „Inline“-Namensräume	104
2.28 Statische Zusicherungen	106
2.29 Expliziter Typkonvertierungsoperator	108
2.30 „alignof“ und „alignas“	110
2.31 „sizeof“ für Elementvariablen	111
2.32 Strukturierte Bindung	112
2.33 Bedingungen mit Initialisierer	114
2.34 Geschachtelte Namensraumdefinition	116
2.35 „Inline“-Variable	117
3 Templates	119
3.1 Variablen-Templates	119
3.2 Typberechnung	121
3.3 Typableitung mit „decltype(auto)“	123
3.4 Unbenannte und lokale Typen als Templateargumente	126
3.5 Default-Argumente für Funktionstemplates	127
3.6 Abgeleiteter Templateparametertyp	128
3.7 Typ-Alias	129
3.8 Externe Templateinstanziierung	131

3.9 Variadische Templates	133
3.10 Faltungsausdrücke	139
3.11 Referenz-Wrapper	141
3.12 Typmerkmale	145
3.13 „forward“-Funktion	156
3.14 „declval“-Funktion	161
3.15 Bedingte Kompilierung im Template	162
4 Bibliothek	165
4.1 Integrale Typen mit definierter Länge	165
4.2 „unique_ptr“-Klasse	168
4.3 „shared_ptr“-Klasse	174
4.4 „weak_ptr“-Klasse	179
4.5 „move“-Funktion	182
4.6 „bind“-Funktion	187
4.7 „function“-Klasse	190
4.8 Einfach verkettete Liste	194
4.9 Array mit fester Länge	197
4.10 Hash-basierte Container	200
4.11 Tupel	207
4.12 Varianten	210
4.13 Beliebige Werte	212
4.14 Optionale Werte	214
4.15 Elementzugriff über den Typ	215
4.16 Brüche	217
4.17 Zeitpunkte und Zeitdauern	220
4.18 Zeitliterale	224
4.19 Zufallszahlen	225
4.20 Reguläre Ausdrücke	230
4.21 „system_error“-Klasse	233
4.22 „next“- und „prev“-Funktionen	237

4.23 Containererweiterungen	238
4.24 IO-Manipulator „quoted“	240
4.25 String-Sichten	241
4.26 Konvertierung von Zeichenketten in Zahlen	243
5 Nebenläufigkeit	245
5.1 „async“-Funktion	245
5.2 Threads	252
5.3 „packaged_task“-Klasse	254
5.4 „promise“-Klasse	257
5.5 „shared_future“-Klasse	262
5.6 Mutexe	264
5.7 Zweistufige Mutexe	267
5.8 „lock_guard“-Klasse	269
5.9 Rekursiver Mutex	272
5.10 „lock“- und „try_lock“-Methoden	276
5.11 „unique_lock“-Klasse	279
5.12 Thread-lokale Daten	282
5.13 Einmalige Ausführung	284
5.14 Bedingungsvariablen	287
5.15 Atomare Operationen	296
5.16 „exception_ptr“-Klasse	302
6 Anhang	305
6.1 Glossar	305
6.2 Literatur	319
Stichwortverzeichnis	321