

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ziel und Zweck des Buches	2
1.2	Vergleich zur Codegenerierung außerhalb von C++	2
1.3	Syntax der verwendeten Beispiele	3
	Literatur	8

Teil I Grundlagen

2	C-Präprozessor	11
2.1	Einbinden von Dateiinhalten	12
2.2	Einfache Makros	12
2.3	Makros mit Parametern	13
2.4	Mehrzeilige Makros	15
2.5	Umwandlung und Bearbeitung von Zeichenketten	16
2.6	Bedingte Ersetzung	17
2.7	Iterationen mit Makros	18
2.7.1	Einfache horizontale Iterationen	18
2.7.2	Iterationen mit #include	20
2.8	Variadische Makros	21
2.8.1	Einführung in variadische Makros	21
2.8.2	Bestimmung der Anzahl der variablen Argumente	23
2.8.3	Iterationen mit variadischen Argumenten	27
2.8.4	Leere Parameter identifizieren	29
2.9	Rechnen mit Makroparametern	30
2.9.1	Basismakros zum Inkrementieren und Dekrementieren	30
2.9.2	Addition und Subtraktion	32
2.9.3	Multiplikation und Division	34
2.9.4	Bestimmung von Primzahlen mit Makros	36
2.10	Compiler-Direktiven	39
	Literatur	40

3	Templates	41
3.1	Funktionstemplates	42
3.1.1	Deklaration	42
3.1.2	Nichttyp-Parameter	44
3.1.3	Reihenfolge der Templateargumente	45
3.1.4	Überladung von Funktionstemplates	45
3.1.5	Vollständige Spezialisierung von Funktionstemplates	47
3.1.6	Rekursiver Aufruf von Funktionstemplates	48
3.1.7	Indirekte partielle Spezialisierung	54
3.2	Klassentemplates	58
3.2.1	Deklaration	58
3.2.2	Nichttyp-Parameter	59
3.2.3	Standardwerte für Templateargumente	60
3.2.4	Vollständige Spezialisierung von Klassentemplates	61
3.2.5	Partielle Spezialisierung von Klassentemplates	61
3.2.6	Rekursiver Aufruf von Klassentemplates	63
3.2.7	Die Verwendung von Enumeratoren	64
3.2.8	Statische If-Bedingung und Switch-Anweisung	65
3.2.9	Der this-Zeiger	69
3.3	Template Template Parameter	70
3.4	Strings als Argumente für Templateparameter	74
3.5	Variadische Templates in C++11	76
3.5.1	Deklaration	76
3.5.2	Variadische Funktionstemplates	77
3.5.3	Variadische Klassentemplates	78
3.5.4	Metaprogrammierung mit variadischen Templates	81
3.6	Das Schlüsselwort constexpr ab C++11	85
	Literatur	87
4	Erweiterte Metaprogrammierung	89
4.1	Typlisten	89
4.1.1	Definition von Typlisten	89
4.1.2	Arbeiten mit Typlisten	93
4.2	Type-Traits	97
4.3	SFINAE	105
4.3.1	Das SFINAE-Prinzip	105
4.3.2	Der sizeof-Trick	107
4.3.3	Gruppierung von überladenen Memberfunktionen	113
4.3.4	Testen auf Software-Updates	115
4.4	Weitere Traits mit sizeof	118
4.5	Assertion zur Kompilationszeit	119
4.6	Umgehen von tiefen Rekursionen	123
4.7	Neue Metafunktionen der STL in C++11 (Type-Traits)	126
	Literatur	131

Teil II Techniken und Anwendungsfälle

5	Sichere Schnittstellen (Interfaces)	135
5.1	Parameterprüfung zur Kompilationszeit	135
5.1.1	Prüfen von Nichttyp-Templateparametern	135
5.1.2	Prüfen von Typ-Parametern	141
5.2	Konfigurationstemplates	144
5.2.1	Konfiguration variabler Typen	145
5.2.2	Konfiguration von veränderlichen Schnittstellen	146
5.2.3	Generierung von Fabrikklassen	148
	Literatur	155
6	C++-Metaprogrammierung für Strukturen	157
6.1	Einführendes Beispiel zur Generierung von Strukturen	157
6.2	Generieren von Strukturen mit Makros	158
6.2.1	Generierung von einfachen Strukturen	158
6.2.2	Erweiterte statische Funktionen	161
6.2.3	Vereinfachungen mit variadischen Makros	164
6.3	Einsatz von Typlisten	164
6.3.1	Anlegen der Typliste	164
6.3.2	Generierung von Strukturen mittels Typlisten	166
6.3.3	Konvertieren von Typlisten	167
6.4	Verwaltungsklassen für Strukturen	170
6.4.1	Klassengeneratoren für Typlisten	170
6.4.2	Basisklasse für alle Typlisten	172
6.4.3	Angepasste Typen für Klassengeneratoren	174
6.4.4	Weitere Optimierung der Strukturdefinition	178
6.4.5	Serialisierung der Struktur	180
6.4.6	Spezialisierung der Verwaltungsklassen	188
6.5	Zugriff auf Elemente der Struktur	189
6.5.1	Zugriff über Index	189
6.5.2	Zugriff über Index zur Kompilationszeit	194
6.5.3	Zugriff über Typnamen	195
6.5.4	Zuweisung mit Kommaoperator	197
6.6	Konvertieren von C++ nach C++/CLI und umgekehrt	200
6.6.1	Konvertieren der Typliste	202
6.6.2	Generieren der C++/CLI-Struktur	206
6.6.3	Kopieren der Daten von C++ nach C++/CLI und umgekehrt	207
6.7	Dokumentation der generierten Strukturen	212
6.7.1	Vorbetrachtungen	212
6.7.2	Erweiterung der Verwaltungsklasse um die Dokumentation	214

6.7.3	Dokumentation mit Typlisten	217
6.7.4	Schreiben der Dokumentation für beliebige Generatoren	225
	Literatur	232
7	Weitere Anwendungsbeispiele für generierte Strukturen	233
7.1	Allgemeine Vorgehensweise	233
7.2	Generierung einer Datenbankschnittstelle	234
7.2.1	Verwaltungsklasse zur Datenbankanbindung	234
7.2.2	Anlegen einer Datenbanktabelle	238
7.2.3	Lesen und Schreiben von Daten	243
7.2.4	Ausblick auf erweiterte Funktionalitäten	248
7.3	Generierung einer Visualisierung	250
7.3.1	Generierung von graphischen Dialogen mit wxWidgets	250
7.3.2	Generierung von graphischen Dialogen mit Qt	255
	Literatur	259
8	Sicheres Rechnen mit Einheiten	261
8.1	Vorbetrachtungen	261
8.2	Einheiten mit Templates realisieren	266
8.3	Das SI-Einheitensystem	269
8.4	Vereinfachte Einheitensysteme	277
8.4.1	Das MKS-Einheitensystem	277
8.4.2	Ein Ingenieur-Einheitensystem	279
8.4.3	Das astronomische Einheitensystem	281
8.5	Konvertierung zwischen den Einheitensystemen	284
8.6	Ein flexibles Einheitensystem	285
8.6.1	Definition des flexiblen Einheitensystems	285
8.6.2	Addition, Subtraktion und Zuweisung	287
8.6.3	Multiplikation und Division	293
8.6.4	Potenz- und Wurzelfunktionen	298
8.6.5	Konvertierung zum SI-Einheitensystem	300
	Literatur	303
9	Speicheroptimierung mit Bitfeldern	305
9.1	Vorbetrachtung	305
9.2	Bitfelder mit Templates realisieren	307
9.2.1	Bitfelder definieren	307
9.2.2	Speichern des Bitfelds	308
9.2.3	Lesen der Elemente	310
9.2.4	Schreiben der Elemente	313
9.2.5	Lesen und Schreiben eines Zeitstempels im Bitfeld	316
9.2.6	Weitere nützliche Methoden für Bitfelder	317
9.3	Umsetzung der Bitfelder mit Typlisten	317
9.3.1	Anlegen der Typliste	318
9.3.2	Definition des Bitfelds	320
9.3.3	Lesen und Schreiben der Elemente	321

9.4	Umsetzung der Bitfelder mit variadischen Templates	324
9.4.1	Definition des Bitfelds	324
9.4.2	Lesen und Schreiben der Elemente	325
9.5	Prüfen von Gleitkommazahlen mit Bitfeldern	327
	Literatur	330
10	Metaprogrammierung mit Gleitkommazahlen	331
10.1	Gleitkommaarithmetik zur Übersetzungszeit	331
10.1.1	Erweiterte mathematische Funktionen	331
10.1.2	Vektor- und Matrizenrechnung	334
10.2	Gleitkommazahlen als Templateargumente	342
10.2.1	Definition und Normalisierung	342
10.2.2	Realisierung der Grundrechenarten	344
10.2.3	Erweiterte mathematische Funktionen	352
10.3	Bruchrechnung mit Templateargumenten	359
10.3.1	Definition und Normalisierung	359
10.3.2	Realisierung der Grundrechenarten	361
10.3.3	Erweiterte mathematische Funktionen	372
	Literatur	376
11	Weitere Beispiele der Metaprogrammierung	377
11.1	Berechnung von Primzahlen	377
11.1.1	Berechnung von kleinen Primzahlen	377
11.1.2	Optimierung des Primzahlentests	379
11.1.3	Berechnung von großen Primzahlen	382
11.2	Berechnung des Osterdatums	385
	Literatur	386
	Stichwortverzeichnis	387