

CARL HANSER VERLAG

Visual C++ .NET
für Einsteiger und Fortgeschrittene

3-446-22329-0

www.hanser.de



Inhalt

Hinweis:

Die Kapitel 3 sowie 27 bis 29 sind aus Platzgründen im Buch nicht abgedruckt, aber auf der beiliegenden CD als PDF-Datei zu finden. Im Inhaltsverzeichnis sind diese Kapitel zusätzlich grau unterlegt.

Einführung	1
1 Einführung	3
1.1 Softwareentwicklung	3
1.1.1 Traditionelle Programmentwicklung	3
1.1.2 Voraussetzungen für die industrielle Softwareproduktion	4
1.1.3 OOA, OOD, OOP	5
1.2 Überblick über die Programmiersprachen	6
1.2.1 Imperative Programmiersprachen	6
1.2.2 Funktionale Programmiersprachen	7
1.2.3 Deklarative Programmiersprachen	7
1.2.4 Objektorientierte Programmiersprachen (OOP)	8
1.2.5 Ereignisorientierte Programmiersprachen	8
1.3 Was heißt eigentlich „objektorientiert“?	9
1.3.1 Objekte	9
1.3.2 Klassen	10
1.3.3 Vererbung	10
1.3.4 Dynamische Bindung (späte Bindung = Late Binding)	11
1.3.5 Geschichte	11
1.3.6 Weitere Details	13
1.3.7 Beispiele	16
1.3.8 Klassenbildung	19
1.3.9 Beispiel	20
1.4 Aufgaben	33
1.5 Installation von Visual Studio .NET	34
1.5.1 Varianten von Visual Studio .NET	34
1.5.2 Grundinstallation	36

1.5.3	Installierbare Komponenten.....	39
1.5.4	Erweiterungsinstallation	42
1.5.5	Funktionstest von Visual Studio .NET	42
1.5.6	Verzeichnisse.....	44
1.6	Allgemeine Hinweise	45
1.6.1	Typografie	45
1.6.2	Begriffe.....	46
1.6.3	Ungarische Notation	47
	Mein erstes Programm.....	51
2	Mein erstes Programm.....	53
2.1	Start der Entwicklungsumgebung.....	53
2.1.1	Neues Projekt anlegen	53
2.1.2	Konsolenanwendung entwickeln	55
2.1.3	Konsolenprogramm testen	62
2.1.4	Fehlersuche	65
2.2	Nützliches zur Oberflächenbedienung.....	68
2.2.1	Maus und/oder Tastatur	69
2.2.2	Desktop.....	70
2.2.3	Explorer	73
2.2.4	Projekte, Dokumente, Fenster und Symbolleisten	74
2.2.5	Fenster, Scheiben.....	75
2.2.6	Symbolleisten	77
2.2.7	Dokumente und Fensterscheiben	80
2.2.8	Visual Studio .NET-Optionen einstellen	82
2.3	MFC-Anwendung.....	83
2.3.1	Anwendungsassistenten aufrufen	83
2.3.2	Rahmengerüst einer MFC-Anwendung	93
2.3.3	Programmierung	100
2.3.4	Programmierungsumgebung	103
2.4	Fehlersuche in Visual C++-Programmen	114
2.4.1	Debugging mit der IDE	115
2.4.2	Zusicherungen (Assertions)	133
2.4.3	Zwischenausgaben.....	135
2.4.4	Aktivierung/Deaktivierung der Fehlerprüfung	136
2.4.5	Präprozessordirektiven	138
2.4.6	Objektreferenzen, Ereignisse und Koordinaten	139
	Einführung in C	5
3	Einführung in C	5
3.1	Allgemeines.....	7
3.2	Struktur von C-Programmen	8
3.3	Daten	11
3.3.1	Daten deklarieren.....	11

3.3.2	Daten zuweisen	16
3.3.3	Daten einlesen bzw. ausgeben	17
3.3.4	Daten mit Funktionen berechnen	17
3.3.5	Daten ansprechen	18
3.3.6	Datentypen wandeln.....	20
3.3.7	Daten im Detail	21
3.3.7.1	Grundlagen (Gleitkommadarstellung nach IEEE 754)	21
3.3.7.2	Wandlungsfehler.....	23
3.3.7.3	Rechenfehler.....	24
3.4	Steuerfluss	27
3.4.1	Grundlagen.....	27
3.4.2	Folge	28
3.4.3	Auswahl	29
3.4.4	Wiederholung.....	33
3.4.4.1	Schleifen	33
3.4.4.2	Abbrüche mit break oder continue.....	36
3.4.4.3	Rekursionen	39
3.4.5	Parallelverarbeitung	41
3.4.6	Schrittweise Verfeinerung	42
3.5	Datenstrukturen	42
3.5.1	Array	43
3.5.2	Zeichenketten (Strings).....	46
3.5.2.1	Zeichenketten speichern	47
3.5.2.2	Zeichenketten definieren	47
3.5.2.3	Zeichenketten ein- und ausgeben.....	48
3.5.2.4	Zeichenketten bearbeiten und vergleichen.....	51
3.5.2.5	Dynamische Strings.....	55
3.5.3	Mehrdimensionale Arrays.....	58
3.5.4	Mehrdimensionale Arrays mit Zeigern	60
3.5.5	Suchen in Arrays.....	63
3.5.6	Arrays sortieren.....	65
3.5.7	Zeiger auf Funktionen.....	67
3.6	C-Systematik	68
3.6.1	Blockbildung.....	69
3.6.2	Speicherklassen.....	70
3.6.3	Operatoren und Ausdrücke	72
3.6.3.1	Rangfolge der Operatoren.....	73
3.6.3.2	Arithmetische Operatoren.....	73
3.6.3.3	Vergleichs- und logische Operatoren	74
3.6.3.4	In- und Dekrement-Operator	74
3.6.3.5	Bitverarbeitung	74
3.6.3.6	ANSI-Escapesequenzen.....	77
3.7	Ein-/Ausgabe-Funktionen.....	77
3.7.1	Konzept der Ströme (Streams).....	78
3.7.2	Dateibearbeitung.....	78

3.7.2.1	Allgemeines.....	78
3.7.2.2	Textdateien.....	80
3.7.2.3	Formatierte Ein- und Ausgabe.....	82
3.7.2.4	Binärdateien.....	84
3.8	Deklaration benutzerspezifischer Datentypen.....	87
3.8.1	Aufzählungen mit enum.....	87
3.8.2	Datenverbunde mit struct.....	88
3.8.3	Überlagerungen mit union.....	91
3.8.3.1	Grundlagen.....	91
3.8.3.2	Hardwarenahe Programmierung mit union.....	91
3.8.3.3	Simulation der OOP in C mit union.....	92
3.9	Präprozessor.....	94
3.9.1	Aufgaben und Funktionen.....	94
3.9.2	Quelltexte einbinden mit #include.....	95
3.9.3	Makros definieren.....	96
3.9.4	Makros mit Parametern.....	97
3.9.5	Bedingte Übersetzung.....	99
3.9.6	Funktionen mit Makros generieren.....	100
3.10	Modularisierung.....	102
3.10.1	Grundlagen.....	102
3.10.2	Modularisierung in C.....	102
3.11	Algorithmen.....	106
3.11.1	Türme von Hanoi.....	106
3.11.2	Formale Sprachen und Automaten.....	109
3.11.2.1	Automaten und reguläre Ausdrücke.....	109
3.11.2.2	Formale Sprachen.....	116
3.11.2.3	Automaten und Graphen.....	121
3.11.2.4	Deterministischer, endlicher Automat (DEA).....	124
3.11.2.5	Reguläre Sprachen und reguläre Ausdrücke.....	126
3.11.2.6	Dualität von Automat und regulärem Ausdruck.....	128
3.11.2.7	Verknüpfung von mehreren endlichen Automaten.....	130
3.11.2.8	Wandlung eines NEA in einen DEA.....	133
3.11.2.9	Begriffe.....	136
3.11.2.10	Reguläre Ausdrücke für endliche Automaten.....	136
3.11.2.11	Operationen auf NEA bzw. DEA.....	137
3.11.2.12	Programmierung.....	139
3.11.2.12.1	Programmarchitektur.....	139
3.11.2.12.2	Alternative Datenstrukturen.....	140
3.11.2.12.3	Simulation eines DEA mit Arrayspeicherung.....	141
3.11.2.12.4	Simulation eines DEA mit Adjazenzlistenspeicherung.....	143
3.11.2.12.5	Myhill-Potenzmengenverfahren.....	150
3.11.2.12.6	Brzozowski-Verfahren.....	152
3.11.2.12.7	Minimieren.....	153
3.11.2.12.8	R^k_{ij} -Verfahren liefert regulären Ausdruck.....	155

3.11.2.12.9	Abschmelzen von Knoten liefert regulären Ausdruck.....	155
3.11.2.12.10	ϵ -Kanten eliminieren	155
3.11.2.12.11	Regulären Ausdruck in NEA transformieren.....	156
3.11.2.12.12	Erweiterte Funktionen	157
3.11.2.12.13	Grammatiken	158
3.11.2.12.14	Mögliche Fortsetzungen	159
	Objektorientiertes Programmieren mit C++	149
4	Objektorientiertes Programmieren mit C++	151
4.1	Klassen und Objekte	151
4.1.1	Einführung	151
4.1.2	Klassische Deklaration neuer Datentypen in C.....	153
4.1.3	Objektorientierte Deklarationen neuer Datentypen in C++	154
4.1.4	Einsatz von Klasse und Objekt	156
4.1.5	Elemente der Klassen.....	158
4.1.6	Auf- und Abbau von Objekten.....	162
4.1.7	Zugriff auf Datenelemente	162
4.1.8	Mehrfache Konstruktoren	165
4.1.9	Quellcode-Browser	169
4.2	Objekte in Objekten.....	172
4.2.1	Objekte als Klasselemente	172
4.2.2	Kopf- und Implementationsdateien.....	175
4.2.3	Konstante Objekte.....	179
4.3	Rückblick.....	182
4.4	Zusammenfassung	183
	Dynamische Speicherverwaltung	185
5	Dynamische Speicherverwaltung.....	187
5.1	Freispeicher (Free Store)	187
5.2	Neue Operatoren zur Objektverwaltung	187
5.2.1	Der Operator <code>new</code>	187
5.2.2	Der Operator <code>delete</code>	188
5.2.3	Speicherbelegung für Standard-Datentypen	189
5.2.4	Arrays	189
5.2.5	Klassen mit Zeigerelementen.....	190
5.2.6	Zuweisungsoperator.....	193
5.3	Der Zeiger <code>this</code>	198
5.3.1	Bedeutung des Zeigers <code>this</code>	198
5.3.2	Funktionsergebnis <code>*this</code>	199
5.4	Zuweisung oder Initialisierung?	200
5.4.1	Unterschiede	200
5.4.2	Kopierkonstruktor	201
5.5	Objekte und Funktionen	203

5.5.1	Werteparameter	203
5.5.2	Objekte als Funktionsergebnisse	204
5.5.3	Objektreferenzen in Funktionen	205
	Weitere Klasseneigenschaften.....	209
6	Weitere Klasseneigenschaften	210
6.1	Statische Elemente	210
6.1.1	Statische Datenelemente.....	210
6.1.2	Statische Methoden.....	212
6.2	Freundschaften	215
6.2.1	Die <code>friend</code> -Funktion	216
6.2.2	Die <code>friend</code> -Klasse.....	217
6.3	Klassenarrays	221
6.3.1	Statische Klassenarrays	221
6.3.2	Dynamische Klassenarrays.....	222
6.4	Weitere Freispeicher-Techniken.....	223
6.4.1	Die Funktion <code>_set_new_handler</code>	223
6.4.2	Überladen der Operatoren <code>new</code> bzw. <code>delete</code>	224
	Vererbung und Polymorphie	225
7	Vererbung und Polymorphie	227
7.1	Verwandte Datentypen in C	227
7.2	Verwandte Datentypen in Visual C++	228
7.2.1	Ableitung einer Unterklasse	228
7.2.2	Konstruktoren abgeleiteter Klassen	231
7.2.3	Redefinition (Überschreiben) von Methoden der Oberklasse.....	233
7.2.4	Erweiterte Zuweisungskompatibilität	236
7.2.5	Oberklassenzeiger für Objektsammlungen	241
7.3	Virtuelle Methoden.....	243
7.3.1	Definition virtueller Methoden	243
7.3.2	Polymorphie	246
7.3.3	Dynamische Bindung	250
7.3.4	Implementation virtueller Methoden	251
7.3.5	Rein virtuelle (abstrakte) Methoden	254
7.3.6	Destruktoren in abgeleiteten Klassen	255
7.4	<code>protected</code> -Klassenelemente.....	256
7.5	Zugriffssteuerung bei der Vererbung	257
7.6	Polymorphie über die Basisklasse	257
7.7	Mehrfachvererbung	262
7.7.1	Grundlagen	262
7.7.2	Virtuelle Klassen	265

Überladen und Typwandlung	267
8 Überladen und Typwandlung	269
8.1 Überladen von Operatoren.....	269
8.1.1 Grundlagen.....	269
8.1.2 Regeln für das Überladen von Operatoren.....	270
8.1.3 Grenzen des Überladens	271
8.1.4 Überladen von Operatoren für eine numerische Klasse.....	271
8.1.5 Überladen mit <i>friend</i> -Funktionen	275
8.1.6 Hinweise zum Überladen arithmetischer Operatoren	277
8.1.7 Überladen von Operatoren für Array-Klassen	277
8.2 Konvertierungen zwischen Klassen.....	283
8.2.1 Konvertierung durch den Konstruktor	284
8.2.2 Konvertierungsoperatoren.....	286
8.2.3 Mehrdeutigkeiten von Konvertierungen	287
Microsoft Foundation Class Library (MFC).....	291
9 Microsoft Foundation Class Library (MFC)	293
9.1 Grundlagen	293
9.1.1 Schlüsselkonzepte.....	295
9.1.2 Arbeiten mit der Programmierumgebung von Visual C++.....	299
9.1.3 Programmvorbereitung	300
9.1.4 Programmgerüst.....	301
9.1.5 Meldungsverarbeitung	307
9.1.6 Ereignis, Meldung und Überschreibung	312
9.1.7 Namens- und Nummerungskonventionen der MFC	314
9.2 Ableiten eigener Klassen von der MFC.....	314
9.2.1 Grundlagen.....	314
9.2.2 Mehrere Dokumenttypen, Ansichten und Rahmenfenster	318
Visual Studio.....	319
10 Visual Studio	321
10.1 Komponenten von Visual Studio.....	321
10.2 MDI-Anwendungsgenerierung mit dem Anwendungsassistenten.....	323
10.2.1 Aufruf	324
10.2.2 Optionen	324
10.2.3 Generierte Klassen	330
10.2.4 Untersuchung der generierten Dateien.....	332
10.2.5 Übersetzen des generierten Programms	336
10.2.6 Programmieren.....	338
10.2.7 Dokument und Ansicht	341
10.2.8 SDI-Anwendungsgenerierung.....	346
10.3 Ressourcendesign mit den Ressourcenassistenten.....	353

10.3.1	Einführung	353
10.3.2	Menügestaltung	356
10.3.3	Zugriffstasten festlegen	360
10.3.4	Zeichenfolgen bearbeiten.....	362
10.3.5	Dialogfeld anlegen.....	364
10.3.6	Dialogklasse anlegen	375
10.3.7	Member-Variablen anlegen	377
10.3.8	Versionsinformationen	382
10.3.9	Anwendungssikone	382
10.3.10	Zusammenfassung	384
10.4	Ereignisfunktionen programmieren.....	385
10.4.1	Ereignisfunktion für eine Menüoption hinzufügen.....	385
10.4.2	Ereignisfunktion programmieren.....	387
10.4.3	Ereignisfunktion für ein Windows-Ereignis hinzufügen	389
10.4.4	Zusammenfassung	396
10.5	Anwendung ohne Dokument/Ansicht-Architektur	398
10.6	Minimale MFC-Anwendung	401
10.6.1	Grundgerüst einer minimalen MFC-Anwendung	401
10.6.2	Fensterklassen und Fensterobjekte	406
10.7	Wie kann ich	408
10.7.1	... erkennen, welche Veränderungen generiert wurden?	408
10.7.2	... verhindern, dass sich der Dialog mit der Zeilenschaltung beendet?	408
10.7.3	... verhindern, dass sich der Dialog mit der Escape-Taste schließt?	412
10.7.4	... mehrere Ereignisse auslösen?	413
	Dialoganwendungen	415
11	Dialoganwendungen	417
11.1	Dialoganwendungen generieren	417
11.2	Programmieren	423
11.3	Ereignisorientierung	434
11.3.1	Benutzereingaben auswerten	434
11.3.2	Erweiterungen programmieren	439
11.3.3	Ereignisbearbeitung verzögern	459
11.4	Menü	462
11.4.1	Menü einbauen	462
11.4.2	Menüoptionen im Dialog bearbeiten	469
11.4.3	Menüoptionen in SDI-/MDI-Anwendungen bearbeiten	470
11.4.4	Kontextmenü	473
11.5	Wie kann ich	478
11.5.1	... eine Dialoganwendung beenden?	478
11.5.2	... einen Dialog zentriert auf dem Bildschirm anzeigen?	478
11.5.3	... ein Steuerelement sichtbar bzw. unsichtbar machen?.....	479
11.5.4	... ein Steuerelement aktivieren bzw. deaktivieren?.....	480
11.5.5	... Daten über ein Steuerelement verarbeiten?	480
11.5.6	... auf ein Dialogfenster bzw. Steuerelemente zeichnen?.....	481

11.5.7	... einen Dialog umkopieren?	483
11.5.8	... die Eingabe auf nationale Eigenheiten umstellen?	484
11.5.9	... Befehlszeilenargumente einlesen?	485
Dialoge und Steuerelemente		489
12	Dialoge und Steuerelemente	491
12.1	Dialoge	491
12.1.1	Allgemeines	491
12.1.2	Meldungsdialoge	492
12.1.3	Hilfsausgaben	497
12.1.4	Standarddialoge	497
12.1.5	Programmierung modaler Dialoge	504
12.1.6	Nichtmodale Dialoge	505
12.1.7	Ansichten und/oder Dialoge	511
12.2	Steuerelemente	513
12.2.1	Probleme	513
12.2.2	Steuerelemente dynamisch anlegen	514
12.2.3	Übersicht über die Standard-Steuerelemente	520
12.2.4	Klasse für eine Dialogressource anlegen	524
12.2.5	Existenz der Steuerelemente	525
12.2.6	Dialog mit der Ansicht verbinden	526
12.2.7	Spezielle Ereignisprogrammierung	526
12.2.8	Wie finde ich die Methoden zum Überschreiben?	527
12.3	Übung zu den Steuerelementen	528
12.3.1	Oberflächenentwurf des Dialogfelds	529
12.3.2	Bezeichnungs- und Textfelder	535
12.3.3	Schaltflächen, Optionfelder und Kontrollkästchen	538
12.3.4	Listen- und Kombinationsfelder	541
12.3.5	Bildfelder	552
12.3.6	Laufleisten, Schieberegler und Drehfeld	556
12.3.7	Fortschrittsanzeige und Zeitgeber	567
12.3.8	Listenelement	572
12.3.9	Strukturansicht (Baumfeld)	581
12.3.10	Register	600
12.3.11	Animationsfeld	602
12.3.12	Zusammenfassung und Problembehandlung	605
12.4	Eigenschaftenfenster	607
12.4.1	Eigenschaftendialog vorbereiten	608
12.4.2	Datumswähler	618
12.4.3	Rich Edit-Steuerelement	621
12.5	Wie kann ich	636
12.5.1	... eine Nachricht ausgeben?	636
12.5.2	... einen Warnton (Pieps) ausgeben?	637
12.5.3	... auf ein Steuerelement zugreifen?	638
12.5.4	... die Farbe eines Steuerelements ändern?	639

12.5.5	... eine Menüoption auf der Oberfläche beeinflussen?.....	645
12.5.6	... die Ein-/Ausgabe und Prüfungen für ein Steuerelement abwandeln?.....	647
12.5.7	... Blasentexte (Tooltips, QuickInfos) anzeigen?.....	651
	ActiveX-Steuerelemente nutzen.....	655
13	ActiveX-Steuerelemente nutzen	657
13.1	Grundlagen.....	657
13.2	ActiveX-Steuerelement im Testcontainer.....	658
13.2.1	Vorbereitung.....	658
13.2.2	Registrierung	659
13.3	Nutzung des Gittersteuerelements	666
13.3.1	ActiveX-Steuerelement in ein Projekt einbinden.....	667
13.3.2	Gittersteuerelement programmieren	673
13.4	Datenübertragung über die serielle Schnittstelle	682
13.4.1	Installation von ActiveX-Steuerelementen.....	682
13.4.2	Steuerelement MSCOMM32.OCX	683
13.4.3	Funktionsweise der seriellen Schnittstelle.....	683
13.4.4	Eigenschaften des Steuerelements MSCOMM32.OCX.....	686
13.4.5	Ereignisse des Steuerelements MSCOMM32.OCX.....	688
13.4.6	Steuerleitungen der Schnittstelle ansprechen.....	688
13.4.7	Schnittstellentest.....	692
13.4.8	Datenaustausch über die serielle Schnittstelle.....	695
13.4.9	Datentyp VARIANT, Klasse ColeVariant	696
13.4.10	Datenaustausch programmieren.....	698
13.5	Wie kann ich	701
13.5.1	... ein „fremdes“ Steuerelement nach Visual C++ transponieren?.....	701
	Symbol- und Statusleisten	705
14	Symbol- und Statusleisten	707
14.1	Symbolleisten.....	707
14.1.1	Eigenschaften der Symbolleiste.....	707
14.1.2	Symbolleisten bearbeiten.....	710
14.2	Statusleiste.....	717
14.2.1	Eigenschaften.....	717
14.2.2	Benutzerdefinierte Statusleiste	720
14.3	Wie kann ich	726
14.3.1	... eine eigene Symbolleiste erstellen?	726
14.3.2	... Steuerelemente in eine Symbolleiste einbauen?	726
	Grafik	727
15	Grafik	729
15.1	Windows Graphical Device Interface (GDI).....	729

15.1.1	Klasse <code>CDC</code>	730
15.1.2	Anzeigekontextklassen <code>CClientDC</code> und <code>CWindowDC</code>	732
15.1.3	Konstruieren und Zerstören von <code>CDC</code> -Objekten	732
15.1.4	Einstellung des Gerätekontexts	734
15.1.5	GDI-Objekte	734
15.1.6	Farbgebung unter Windows	738
15.1.7	Koordinatensystem	742
15.1.8	Zeichenfunktionen	745
15.1.9	Schriftarten	747
15.2	Übungen mit dem GDI	749
15.2.1	Freihandzeichnen	749
15.2.2	Zeichnung beschneiden (Clipping)	754
15.2.3	Zeichnungen persistent machen	758
15.2.4	Interaktiv zeichnen	761
15.2.5	Ziehen und Ablegen (Drag and Drop)	764
15.2.6	Grafikfenster mit Bildlaufleisten	770
15.2.7	Schriften	774
15.3	Mathematische Funktionen	782
15.3.1	Mathematische Grundlagen	782
15.3.2	Interpolation mit stückweise definierten Funktionen	783
15.3.3	Realisationen	810
15.4	Bitmaps	829
15.4.1	GDI-Bitmaps und DIBs	829
15.4.2	Einsatz von GDI-Bitmaps	829
15.4.3	Bitmaps als Ressource anzeigen	830
15.5	Darstellung geräteunabhängiger Bitmaps (DIBs)	837
15.5.1	Klasse <code>CDib</code>	837
15.5.2	PCX-Format für Bilddateien	838
15.5.3	Programmierung	840
15.6	Dateiformate	864
15.6.1	Einführung	864
15.6.2	BMP-Format	871
15.7	Wie kann ich	873
15.7.1	... die Schrifteigenschaften zur Laufzeit einstellen?	873
15.7.2	... Spezialeffekte in Schriften erzeugen?	874
15.7.3	... einen Funktionator schreiben?	874
15.7.4	... einen Interpolator schreiben?	875
15.7.5	... einen Approximator schreiben?	876
15.7.6	... einen Fraktator schreiben?	877
15.7.7	... ein CAD-Programm schreiben?	879
	Dokument und Ansicht, Auflistungen	881
16	Dokument und Ansicht, Auflistungen	883
16.1	Hauptrahmenfenster	883
16.1.1	Kindfenster des Hauptrahmenfensters	883

16.1.2	Meldungsverarbeitung	883
16.1.3	Aktualisierung von Menüoptionen	886
16.2	Interaktionen zwischen Dokument und Ansicht	886
16.2.1	Methoden der Klasse <code>CView</code>	887
16.2.2	Methoden der Klasse <code>CDocument</code>	888
16.2.3	Reihenfolge der Ereignisse	891
16.3	Ausgewählte Klassen der MFC	892
16.3.1	Klasse <code>CFormView</code>	892
16.3.2	Klasse <code>CObject</code>	893
16.4	Auflistungen	897
16.4.1	Grundlagen	897
16.4.2	Typsichere Auflistungen	900
16.5	Übungen	910
16.5.1	MFC-gerechte Datenklassen	910
16.5.2	Listenverarbeitung mit Personen <code>CPersonInfo</code>	927
16.5.3	Listenverarbeitung mit Typzeigern	927
16.6	Wie kann ich	938
16.6.1	... eine Liste von Punkten <code>CPoint</code> verwalten?	938
	Serialisierung und Persistenz	939
17	Serialisierung und Persistenz	940
17.1	Grundlagen der Serialisierung	940
17.2	Klassen <code>CFile</code> und <code>CArchive</code>	941
17.3	Entwurf einer serialisierbaren Klasse	941
17.3.1	Benutzerdefinierte Methode <code>Serialize</code>	941
17.3.2	Eingebettete Objekte und Zeiger	943
17.3.3	Serialisierung von Auflistungen (Collections)	945
17.3.4	Aufruf der Methode <code>Serialize</code>	945
17.4	Persistenz über die Registrierung	946
17.4.1	Registrierdatenbank	946
17.4.2	Zugriffsmethoden auf die Registrierdatenbank	947
17.5	Übungen	948
17.5.1	Serialisierung der Personalliste	948
17.5.2	Persistente Fenster	957
17.5.3	Persistente Personalliste	963
17.5.4	Serialisierung und Dateiverarbeitung	965
17.5.5	Textdateien	969
17.6	Wie kann ich	970
17.6.1	... eine erfolgreiche Serialisierung sicherstellen?	970
17.6.2	... eine Serialisierungsdatei untersuchen?	970
	MDI- und (fast) gleichberechtigte Fenster	971
18	MDI- und (fast) gleichberechtigte Fenster	973
18.1	Grundlagen der MDI-Anwendungen	973

18.1.1	Begriffe	973
18.1.2	MDI-Anwendungsobjekt	975
18.1.3	MDI-Dokumentvorlage.....	975
18.2	MDI-Rahmen- und Kindfenster.....	976
18.2.1	Anwendungsrahmen	976
18.2.2	Ressourcen	977
18.2.3	Menüs	977
18.3	Übungen	978
18.3.1	Umwandlung einer SDI- in eine MDI-Anwendung.....	978
18.3.2	Fenster auf höchster Ebene	988
18.4	Wie kann ich	991
18.4.1	... eine MDI-Anwendung ohne Kindfenster öffnen?.....	991
18.4.2	... Kindfenster ordentlich anordnen?.....	992
Erweiterte Ansichten.....		993
19	Erweiterte Ansichten	994
19.1	Teilbare Fenster	994
19.1.1	Grundlagen.....	994
19.1.2	Vorbereitete Methoden	995
19.2	Übungen zu SDI-Anwendungen.....	997
19.2.1	SDI-Anwendung mit dynamischer Teilung, gleiche Ansichten.....	997
19.2.2	SDI-Anwendung mit unterschiedlichen Ansichten.....	1003
19.2.3	Umschalten von Ansichtsklassen in einer SDI-Anwendung.....	1008
19.3	Wie kann ich	1011
19.3.1	... ein mehrstufig geteiltes Fenster erzeugen (Fenster im Fenster)?	1011
19.3.2	... dynamisch geteilte Fenster verarbeiten?	1014
Softwaremodule.....		1017
20	Softwaremodule	1018
20.1	Grundlagen	1018
20.2	Erstellen einer .LIB-Bibliothek.....	1019
20.2.1	Vorhandene Klassen in eine .LIB-Bibliothek integrieren	1019
20.2.2	Statische Bibliothek nutzen.....	1022
Dynamic Link Libraries		1027
21	Dynamic Link Libraries	1029
21.1	Grundlagen	1029
21.1.1	Architektur.....	1029
21.1.2	Struktur einer DLL.....	1033
21.1.3	Deklaration exportierter DLL-Funktionen.....	1034
21.1.4	Suchhierarchie des aufrufenden Programms.....	1035

21.1.5	Reguläre DLLs und MFC-Erweiterungs-DLLs	1035
21.2	Erstellen einer regulären DLL	1036
21.2.1	DLL erstellen	1036
21.2.2	DLL unter Visual Basic testen	1042
21.2.3	DLL unter Visual C++ testen	1049
21.2.4	Probleme der Parameterübergabe	1053
21.3	MFC-Erweiterungs-DLL	1064
21.3.1	Klasse CFestestFenster	1064
21.3.2	Erzeugung der DLL	1070
21.3.3	Testprogramm für eine DLL	1072
21.4	DLL-Spion	1075
21.4.1	PE/COFF-Dateiformat	1075
21.4.2	Datei-Header	1077
	ActiveX-Steuerelemente	1081
22	ActiveX-Steuerelemente	1083
22.1	Grundlagen	1083
22.1.1	Hauptkomponenten eines ActiveX-Steuerelements	1084
22.1.2	Steuerelemente mit Fenstern und ActiveX-Steuerelement-Container	1084
22.1.3	Aktive und inaktive Zustände eines ActiveX-Steuerelements	1085
22.1.4	Serialisieren	1086
22.1.5	Unterschiede zwischen ActiveX- und normalen Steuerelementen	1086
22.2	ActiveX-Steuerelemente erstellen	1087
22.2.1	Grundstruktur erstellen	1087
22.2.2	Programmieren eines ActiveX-Steuerelements	1098
22.2.3	Benutzerdefinierte Eigenschaft hinzufügen	1106
22.2.4	Hinzufügen einer benutzerdefinierten Get/Set-Eigenschaft	1113
22.2.5	Ereignisse programmieren	1117
22.2.6	Verarbeiten von Text und Schriftarten	1134
22.2.7	Eigenschaftenseiten programmieren	1140
22.2.8	ActiveX-Steuerelement unter Visual Basic testen	1148
22.2.9	Einfache Datenverbindung	1155
22.2.10	Versionierung und Serialisierung	1163
	Multitasking/Multithreading	1173
23	Multitasking/Multithreading	1175
23.1	Grundlagen	1175
23.1.1	Multitasking	1175
23.1.2	Multithreading	1176
23.2	Programmierung paralleler Prozesse	1177
23.2.1	Multitasking	1177
23.2.2	Multithreading	1177
23.3	Beispiele	1194
23.3.1	Multitasking in einer dialogfeldbasierenden Anwendung	1194

23.3.2	Multitasking in einer SDI-/MDI-Anwendung.....	1194
23.3.3	Multithreading	1200
23.4	Wie kann ich	1210
23.4.1	... feststellen, ob ein Faden noch aktiv ist?.....	1210
23.4.2	... weitere Fäden mit eigenen Fenstern starten?	1212
23.4.3	... mehrere Fäden ordnungsgemäß beenden?	1217
23.4.4	... mehrere Arbeiterfäden verarbeiten?.....	1222
23.4.5	... gleichberechtigte Fäden untereinander kommunizieren lassen?	1223
	Internet/Intranet/lokale Netzwerke.....	1229
24	Internet/Intranet/lokale Netzwerke	1230
24.1	Grundlagen	1230
24.1.1	Das ISO-Referenzmodell.....	1230
24.1.2	TCP/IP	1234
24.1.3	Adressierung	1237
24.1.4	Protokolle höherer Ebenen.....	1239
24.2	Sockets.....	1242
24.2.1	Einführung	1242
24.2.2	Socket-Modi.....	1243
24.3	Socket-Programmierung mit der MFC	1244
24.3.1	Die Klasse CAsyncSocket.....	1245
24.3.2	Die Klasse CSocket.....	1248
24.4	Übungen	1249
24.4.1	Netzwerkanwendung.....	1249
24.4.2	Datensauger	1261
	Datenbankzugriff	1273
25	Datenbankzugriff	1275
25.1	Grundlagen	1275
25.1.1	Microsoft Datenbankkomponenten.....	1275
25.1.2	Architektur einer Microsoft-Datenbankanwendung.....	1277
25.2	Datenbanken und/oder Serialisierung?	1279
25.3	Datenbankentwurf	1281
25.3.1	Theorie.....	1281
25.3.2	Das Datenbankbeispiel.....	1283
25.4	Open Database Connectivity (ODBC).....	1288
25.4.1	Grundlagen.....	1288
25.4.2	Registrierung von ODBC/OLE DB-Datenbanken	1288
25.4.3	Direkter Zugriff auf die ODBC-Schnittstelle.....	1292
25.5	DAO (Data Access Objects).....	1293
25.6	ADO (ActiveX Data Objects) und ADO.NET	1294
25.6.1	Architektur von ADO	1294
25.6.2	Klassen-/Objektmodell von ADO.....	1296
25.6.3	Auflistungen unter ADO.....	1298

25.6.4	Ereignisprozeduren unter ADO	1300
25.6.5	Datenbankzugriff	1301
25.6.6	ADO und COM	1302
25.7	Anwendungsbeispiele zur DAO	1307
25.7.1	Grundprogramm	1307
25.7.2	Sortierung	1314
25.7.3	Filterung	1315
25.7.4	Neueingeben und Löschen von Datensätzen	1317
25.8	Anwendungsbeispiele zur ADO	1318
25.8.1	Grundprogramm	1318
25.8.2	Verwenden der #import-Compilerdirektive.....	1319
25.8.3	Verwenden von Property-Direktiven.....	1323
25.8.4	Auflistungen, GetItem-Methode und die Item-Eigenschaft	1324
25.8.5	COM-spezifische Datentypen.....	1325
25.8.6	Fehlende Parameter und Standardparameter	1327
25.8.7	Fehlerbehandlung	1328
25.8.8	Visual C++-Äquivalente zu den Visual Basic-Konventionen	1328
25.9	Visual C++-Erweiterungen für ADO	1338
25.9.1	Verwenden der Visual C++-Erweiterungen für ADO	1338
25.9.2	ADO mit Visual C++-Erweiterungen (Beispiel)	1345
25.10	Verwenden der ADO-Datenbindung in VC++	1348
25.10.1	Programmierter ADO-Zugriff	1348
25.10.2	ADO-Datenbindung in Visual C++	1361
	Verwalteter Code im .NET Framework	1367
26	Verwalteter Code im .NET Framework	1369
26.1	Grundlagen des .NET Frameworks	1369
26.2	Verwaltete C++-Anwendung erstellen	1372
26.3	Windows Form-Anwendung programmieren	1374
26.4	Bestehende MFC-Anwendung mit verwalteten Erweiterungen	1378
26.5	Steuerelemente und Grafik	1380

Druckfunktionen 3

27	Druckfunktionen	3
27.1	Grundlagen	3
27.1.1	Standarddialoge	3
27.1.2	Seitenauswahl	5
27.1.3	Unterschiede zwischen angezeigten und gedruckten Seiten	6
27.2	Druckvorschau.....	6
27.3	Druckerprogrammierung	6
27.3.1	Druckerkontext	7
27.3.2	Ereignisfunktion CView::OnPrint	9
27.3.3	Druckbeginn und Druckende	10
27.4	Beispiele	11

27.4.1	WYSIWYG-Druckausgabe.....	11
27.4.2	Mehrseitiger Druck	15
Kontextabhängige Hilfe		3
28	Kontextabhängige Hilfe	3
28.1	Grundlagen	3
28.2	Help Workshop zur Erstellung einer Hilfe	4
28.2.1	Einführung	4
28.2.2	Rich Text Format (RTF)	8
28.2.3	Interne Logik einer Hilfedatei	10
28.2.4	Klassische Hilfe	11
28.2.5	Hilfe im Win32-Stil	18
28.3	HTML Help Workshop zur Erstellung einer Hilfe	23
Fehlersuche und Ausnahmebehandlung.....		3
29	Fehlersuche und Ausnahmebehandlung	3
29.1	Debugmodus und Auslieferungsmodus	3
29.2	Compilerfehler.....	5
29.2.1	Unerwartete Fehlerstelle	5
29.2.2	Fehlende Semikola.....	7
29.2.3	Ignorierte oder nicht gefundene Funktionsaufrufe.....	8
29.3	Fehlersuche im Debugmodus	11
29.3.1	Fehlersuche mit der MFC	12
29.3.2	Compilervariable _DEBUG.....	12
29.3.3	Zwischenausgaben	13
29.3.3.1	Zwischenausgaben mit dem TRACE-Makro	13
29.3.3.2	Zwischenausgaben mit dem ATLTRACE2-Makro	14
29.3.3.3	Ausgabe des Inhalts eines Objektes mit DUMP	15
29.3.4	Zusicherungen.....	16
29.3.4.1	Zusicherungen mit ASSERT	16
29.3.4.2	Zusicherungen mit ASSERT_VALID	22
29.3.4.3	Zusicherungen mit VERIFY	24
29.3.5	Speicherüberwachung	25
29.3.5.1	Verfolgen der Speicherreservierung	27
29.3.5.2	Feststellen von Speicherverlusten.....	27
29.3.5.3	Speicherdiagnose	28
29.3.5.4	Eingrenzen von Speicherverlusten.....	29
29.3.5.5	Ausgeben der Speicherstatistik	30
29.3.5.6	Ausgabe aller Objekte.....	32
29.4	Automatische Überprüfungen.....	33
29.4.1	Undefinierte Variablen.....	33
29.4.2	Undefinierte Zeiger	34
29.4.3	Zerstörung des Kellerspeichers (Stacks).....	35

29.4.4	Zerstörung des Heaps	36
29.4.5	Zahlenüber- und -unterläufe	36
29.4.6	Überläufe des Stacks bzw. Heaps	38
29.5	Manuelle Fehlersuche.....	39
29.6	Ausnahmebehandlung	40
29.6.1	Grundlagen	40
29.6.2	Interne Ausnahmen auslösen	42
29.6.3	Benutzerdefinierte Ausnahmebehandlung	45
29.7	Ausnahmebehandlung in verwaltetem C++	47
29.7.1	Ausnahmen abfangen	47
29.7.2	Ausnahmen aufwerfen	49
29.7.3	Ausnahmen im Konstruktor verwenden	50