

*Michael Hyman &
Bob Arnson*

Visual C++ 6 für Dummies

SONDERAUSGABE

*Übersetzung aus dem
Amerikanischen von
Barbara und Uwe Jaekel*



WILEY-VCH Verlag GmbH & Co. KGaA

Inhaltsverzeichnis

Einführung	23
Ist das jetzt ein Buch über C++ oder Visual C++?	23
Warum ein Buch für Dummies?	24
Wer dieses Buch benutzen sollte	24
Wie dieses Buch aufgebaut ist	25
Über die CD	25
Wie man dieses Buch benutzt	25
Icons, die in diesem Buch benutzt werden	26
Warum heißt es C++?	26
Teil 1	
Visual C++ in neun einfachen Kapiteln	27
Kapitel 2	
Was gehört zum Visual C++-Paket?	29
Die Visual C++-Features	29
Ein Compiler für 1001 Programme	30
Die Venusfliegenfalle der Programmentwicklung: Der Debugger	30
Oje, Dialoge und Menüs und Bitmaps und ...	31
1, 2, 3, 4, heute integrieren wir	32
Ein solides Grundgerüst für den Programmaufbau	32
Bibliotheken, die niemals Überziehungsgebühren verlangen	34
Maxwell Smart-Utilities	34
Beispielprogramme für den Anfang	34
Kapitel 2	
Windows-Programme aus dem Hut zaubern	37
Visual C++ anwerfen und kennenlernen	37
Die grundlegenden Strukturen von Visual C++	38
Werkzeuge für den Umgang mit den Visual C++-Strukturen	38
Eine Windows-Anwendung kreieren	39
Los geht's - Wo steckt der Assistent?	40
Alles fest im Griff mit den Weiteren Optionen	45
Ich will einen fesselnden Titel!	47
Jenseits des Anwendungsassistenten: Textdateien und Code	48

Das Codieren möge beginnen	49
Lauf, Programm, lauf	50
Was bei Fehlern zu tun ist oder wenn das Kompilieren ewig dauert	53

Kapitel 3***Der Assistent, ein wahrer Zauberer*****55**

Passen Sie Ihr Programm an	55
Der Assistent kümmert sich um die Klassenarbeit	56
Anpassen mit dem Dialog-Editor	57
Fügen Sie etwas ganz Neues hinzu: eine Schaltfläche	59
Der Beweis? Es läuft!	61
Sitzen Sie nicht einfach nur auf Ihrer Schaltfläche herum	61
Erzeugen Sie ein neues Dialogfeld	65
Geben Sie Ihrem Dialogfeld mit dem Klassenassistenten eine gewisse Klasse	66
Neue Menüpunkte hinzufügen	69
Fügen Sie alles zusammen: Das neue Dialogfeld einschließen	71
Nochmali!	74

Kapitel 4***Ich arbeite da gerade an einem neuen Projekt, n'est-ce pas?*****75**

Ich brauche keine Millionen, ich brauche nur ein Projekt	75
Projektdateien machen Ihr Leben leichter	76
Arbeitsbereiche und Projekte	76
Ein neues Projekt erstellen	77
Die Details des neuen Projekts festlegen	77
Dateien zum Projekt hinzufügen	78
Das Projektfenster zum Hof	80

Kapitel 5***Alles, was Sie brauchen, ist ein guter Editor*****83**

Ein Code, der nicht schmutzt	83
Der Editor-Zeremonienmeister	85
Und das alles live und in Farbe	87
Online-Hilfe: F1! F1! Hört mich denn niemand?	89
Suchet, so werdet Ihr finden	89
Reguläre Ausdrucksstärke	92

Kapitel 6	
Kompilier's noch einmal, Sam	93
Einmal ist keinmal	93
Aus Syntaxfehlern lernen	94
Schlagen Sie Warnungen nicht einfach in die Windows	96
Warum korrigiert der Compiler keine Fehler?	96
Es gibt mehr als eine Art zu kompilieren	97
Kapitel 7	
Eine Debuggerfahrt durch die Eifel	99
Was ist der Unterschied zwischen einem Syntaxfehler und einem Bug?	100
Ein Überblick über den Debugging-Prozess	100
Wer schaut denn da aus dem Fenster?	102
Haltet den Fehler!	102
Ein kleiner Schritt für einen Debugger ...	103
Der große Debugger beobachtet Dich	105
Deine Werte sind von Übel	105
Sie sind in Eile? Kein Problem!	106
Sie haben es aber noch eiliger?	106
Bugs Bunnys Live-Vorstellung	108
Bühne frei für den Bug	108
Ein Haufen Fehler	109
Wo liegt der Bug im Pfeffer?	109
Sind wir schon da?	112
Aber es gibt weitere Bugs	113
Noch einmal zum Mittippen	114
Beim Jupiter, Zenturio, es funktioniert	114
Nur noch aufräumen, und wir sind fertig	115
Der Debugging-Blues	116
Kapitel 8	
Programmieren in Saus und Browse	119
Ich soll mein Projekt browsen!?	119
Was bedeuten diese Hieroglyphen?	122
Die meistgenutzten Browser	123
Legen Sie den Browser in Reichweite ab	123
Vorbereiten des Browsers	124
Durchsuchen Sie Ihren Stammbaum	124
Die Suche eingrenzen	125
Manchmal funktioniert etwas nicht	126

Kapitel 9		
Erwägen Sie alle Optionen		129
Gesagt, getan		129
Änderungen für einzelne Dateien		131
Header-Dateien ändern		132
Optionen einstellen in der Visual C++-Umgebung		133
Die Registerkarte Verzeichnisse		133
Die Registerkarte Format		134
Die Registerkarte Symbolleisten		134
Teil II		
Was Sie schon immer über C++ wissen wollten, aber nie zu fragen wagten		137
Kapitel 10		
Dein Programm, das unbekannte Wesen		139
Alle Programme starten in der Main Street		141
Schau doch mal in der Bibliothek nach		142
Lösen Sie Ihre Probleme mit einem Programm		142
Das Jukebox-Programm lesen		143
Kapitel 11		
Einführung in die objektorientierte Programmierung		145
Wie funktioniert objektorientierte Programmierung?		145
Das Objekt ist das Objekt		146
Kapitel 12		
Teile und herrsche		151
Was schreibe ich bloß in eine Quellcode-Datei?		152
Fangen Sie ganz vorne an		153
Lies mich, fühl mich, berühr mich, schreib mich		154
Wo geht's denn hier zur nächsten Zeile?		156
Cin Cin		156
Der Kommentar		157
Leihen Sie sich Funktionen in der Bibliothek aus		158
Endlich: Hello World. Ist da draußen jemand?		159
Alles schön der Reihe nach		159

Kapitel 13	
Daten sind schon komische Typen	163
Feierliche Deklaration einer Variablen	164
Die elementaren Datentypen	165
Weniger gebräuchliche Datentypen	165
Typsicherheit	166
Manche Dinge ändern sich nie: Konstanten	168
Geben Sie Ihrem Programm etwas Konstantes	169
Konstanten und Fehlervermeidung	170
Kapitel 14	
Deine Variable, das wandelbare Wesen	171
Variablen benennen	171
Variablen definieren	173
Variablen initialisieren	174
Kapitel 15	
Strukturen: Baumaterial für Variablen	175
Strukturen deklarieren	175
Strukturen bei der Arbeit	176
Große Strukturen aus kleinen Strukturen aufbauen	178
Geben Sie Ihrem Programm Struktur(en)	179
Kapitel 16	
Wie Sie einen guten Ausdruck hinterlassen	181
Smooth Operators	182
Komplexe Operatoren	182
Der <code>++</code> -Operator	183
Der <code>>></code> -Operator	184
Der <code><<</code> -Operator	184
Echt bool, ey	185
Der Zuweisungsoperator	187
Komplexe Ausdrücke	189
Gibt es hier keine Abkürzung?	189
Bit' schön: Noch mehr Operatoren	191
Wenn das Wörtchen if nicht wär ...	193
Ich rufe diese Operatoren zur Ordnung	194
Aber jetzt ein paar Beispiele	195

Kapitel 17		
Fluss ohne Wiederkehr?		197
Die großen drei: if, for und while	198	
Wenn ich einmal reich wär': Die if-Anweisung	198	
An und für sich könnte es so weitergehen	201	
Warte, warte noch ein while-chen	204	
Noch mehr Fluss im Überfluss	205	
Die Schlüsselwörter switch, case und break	205	
Das Schlüsselwort do (wah diddy diddy dum diddy do)	207	
Gehen Sie in das Programmierer-Gefängnis, und nehmen Sie das goto gleich mit	207	
Kapitel 18		
Eine verbesserte Jukebox		209
Die Anatomie des neuen Jukebox-Programms	209	
Definition der Konstanten	209	
Deklaration der Struktur	210	
Die do-Schleife	210	
Die switch-Anweisung	211	
Die cout-Zeile	211	
Kapitel 19		
Und welche Funktion haben Sie?		213
Gleich am Anfang ein paar Anweisungen	213	
Wie man eine Funktion bastelt	214	
Argumente (Ja, Nein, Ja, Nein)	215	
Funktionen, die Werte zurückliefern	216	
Noch einmal zur Fakultät	219	
Wie man Programme liest, die Funktionen enthalten	221	
Speichern von Informationen	221	
Rekursion ... Rekursion ... Rekursion ...	223	
Jetzt noch ein paar Pünktchen	226	
Immer diese Standardargumente	226	
Kapitel 20		
Zeiger mir den richtigen Weg		227
Wie man Zeiger benutzt	227	
Sie haben sie schon benutzt	228	
Zeiger und Variablen	228	

Zeiger gehen mir auf den Zeiger	229
Zeig' er mir ein Beispiel	230
Wie man Zeiger in C++ verwendet	233
Geben Sie Ihrem Zeiger eine Adresse	233
Zeigen, Lesen, Ändern	235
Ändern von Werten in einer Struktur	235
Zeig' doch mal eben	236
Mit new neuen Speicherplatz reservieren	236
Ein klassisches Programm: Verkettete Liste	238
Endlich frei	241
Zeigen Sie auf die Null	242
Fehlanzeiger	242
Stringsda	244
Treiben Sie es nicht zu void*	245
Einige einfache Zeiger	247
Zeigen Sie auf dieses Kapitel	247
 Kapitel 21	
Eine noch lautere Jukebox	249
Wie das verbesserte Jukebox-Programm funktioniert	249
 Kapitel 22	
Felder, Wiesen und Aufzählungen ...	251
Wie verwende ich enums?	251
Gehen Sie auf Nummer Sicher	252
cinderei	253
Auf, auf ins Feld	253
Greifen Sie zu!	254
Felder initialisieren	255
Strings und Felder	255
Dr. Lizardo-Special: Mehrdimensionale Felder	257
Dies sind keine gewöhnlichen Tipps; es sind mehrdimensionale Tipps	258
 Kapitel 23	
Ein kurzer Versuch über die Jukebox	259
Änderungen, die für die Verwendung von Feldern nötig sind	259
Weitere Änderungen am Programm	260

Kapitel 24	
Der Gültigkeitsbereich von Namen	263
Aber wen interessiert das?	264
Regeln für Gültigkeitsbereiche: Wann ist EineRose nicht EineRose?	266
Wie Sie den Überblick nicht verlieren	267
Vom Tellerwäscher zum Programmierer	267
Teil III	
Und jetzt zu etwas völlig Objektorientiertem	271
Kapitel 25	
Eine Klasse für sich	273
Willkommen im Klassenraum 101	274
Datenelemente	274
Elementfunktionen	275
Eine Klasse deklarieren	275
Eingeschränkter Zugriff	275
Geschützter Zugriff	276
Wie man schreibgeschützte Variablen einführt	277
Elementfunktionen definieren	277
Und wie arbeitet man nun mit Klassen?	278
Auf Klassenelemente zugreifen	278
Mit Elementfunktionen auf Elemente zugreifen	280
Geh' zum Klassensprecher	280
Objektorientiertes Nachdenken über Musik	280
Zeit für eine Klassenarbeit: Was haben wir gelernt?	282
Kapitel 26	
Boa Konstruktor und Boa Destruktor	285
Wenn der Konstruktor nicht da ist, tanzen die Wanzen	286
Mehr Spaß mit mehreren Konstruktoren	287
Sie müssen schon gute Argumente haben	288
Das öffentliche und das Privatleben von Konstruktoren	288
Räum' Deinen Mist doch selber weg	289
Denken Sie auch an den dynamischen Kram	290
Was passiert, wenn ich eine Klasse in einer Klasse habe?	290
Wie man ein objektorientiertes Programm liest	292

Kapitel 27		
Jukebox++		295
Die Klassen der Jukebox		295
Und was passiert dann?		298
Wie die verkettete Liste funktioniert		300
Alles fließt		301
Kapitel 28		
Ein Vermögen erben		303
Wie Vererbung funktioniert		303
Der Einfluss von public, private und protected		305
Überschreiben Sie Ihr Vermögen		305
Wie man auf Funktionen der Basisklasse zugreift		306
Ein Beispielprogramm		307
Wie Vererbung auf Konstruktoren und Destruktoren wirkt		309
Zeiger und abgeleitete Klassen		310
Einführung in das Erbrecht für Programmierer		311
Kapitel 29		
Diese virtuellen virtuellen Funktionen		313
Wenn ich niemals Zeiger auf Objekte benutze, kann ich dann dieses Kapitel überspringen?		315
Wie Sie erkennen, ob Sie eine virtuelle Funktion benötigen		315
Eine virtuellen Funktion deklarieren		316
Glauben Sie nichts, was Sie nicht selbst kompiliert haben		317
Richten Sie sich auf den Nachwuchs ein		321
Kapitel 30		
Shirley Templates		323
Schablonen, die sich lohnen		323
Schablonenhafte Programmierung		325
Elementfunktionen für eine Schablone definieren		326
Eine Schablone verwenden		326
Regeln für die Verwendung von Zeigern mit generischen Klassen		327
Die Verwirrung mindern		327
Packen Sie es in eine Header-Datei		328
Ein Programm mit einer Schablone für verkettete Listen		329
Der Code		331

Kapitel 31		
Weiter geht's mit Musik		333
Wie man ein Programm mit mehreren Dateien liest		334
Was in diesem Programm geschieht		334
Die FancyJukebox-Klasse		335
Die PreviewSongList-Klasse		336
Vermischen der Objekte in der verketteten Liste		337
Stecken Sie Klassendeklarationen in Header-Dateien		339
Endlich, das Beispielprogramm		341
Kapitel 32		
Ein- und Ausgabe		343
Dateien und Streams		343
Wie man mit einer Variablen auf mehrere Dateien zugreift		344
Fünf Fakten über Files		344
Einlesen von Zahlen und Wörtern in Dateien		345
Der Beispiel-Code		347
Einige besondere Dinge, die man in einen Stream werfen kann		349
Fill und Width für Integer setzen		349
Kapitel 33		
Ausnahmen bestätigen den Handle		351
Fehlerbehandlung auf die alte Art		351
Fehlerbehandlung auf die neue, verbesserte Art		353
Dr. Seltsams Ausnahmebehandlungszimmer		353
Ein ausnehmend gut behandeltes Beispiel		353
Hans Dampf in allen Sackgassen		355
Ich vergesse nie ein Gesicht, aber in Ihrem Fall werde ich eine Ausnahme werfen		357
Syntax der Ausnahmebehandlung		358
Es sieht gut aus, aber dieses Zeug ist ziemlich verwirrend		360
Schauen Sie sich ein Beispiel an		362
Von Fehlerbehandlungsklassen erben		364
Fünf Regeln für Ausnahmefolge		364
Kapitel 34		
Eine Jukebox für Ausnahmekünstler		365
Die Eingabedateien		365
Wie die aktualisierte Jukebox funktioniert		366
Einlesen von Dateien durch Streams		366
Die Ausnahmebehandlung		367

Kapitel 35		
Überladen ohne Schaden		369
Überladen einer Elementfunktion		369
Überladen einer vordefinierten Funktion		371
Diese Operatoren sind überladen, Captain		371
Warnung: Nun wird's wirklich kompliziert		372
Überladen innerhalb und außerhalb von Klassen		373
Einladen, überladen, ausladen		374
Was Sie nicht tun können		377
Freunde in der Programmiernot		377
Wofür Freunde gut sind		377
Wie Sie sich Freunde machen und damit Programme beeinflussen können		378
Kapitel 36		
Die Jukebox der Pandora überladen		379
Wie es funktioniert		379
Der Code		381
Teil IV		
Zehn, die die Welt bewegen		383
Kapitel 37		
Zehn Syntaxfehler		385
Falsche Include-Pfade		385
Fehlendes ;		386
Das Laden einer Header-Datei vergessen		386
Die Aktualisierung einer Klassendeklaration vergessen		387
Verwendung des Klassennamens anstelle des Variablenamens		387
; hinter einer Klassendeklaration vergessen		388
Das Schlüsselwort public in einer Klassendefinition vergessen		388
Verwendung eines falschen Variablenamens (weil man den richtigen peinlicherweise vergessen hat)		389
Verwendung von -> statt . und umgekehrt		389
Eine fehlende }		390

Kapitel 38		
Zehn weitere Syntaxfehler		391
Beenden eines Kommentars vergessen		391
Verwendung eines falschen Typs für eine Variable		392
Es lief wunderbar, als es noch ein C-Programm war, aber jetzt wird es nicht kompiliert		393
Nichts anstelle von void setzen		393
Definition einer Template-Elementfunktion in einer Header-Datei vergessen		394
Einen dringend benötigten öffentlichen Konstruktor vergessen		394
Ein ; am Ende von #define		394
Das Erstellen einer Projektdatei vergessen		395
Plattenspeicher voll		395
Plötzlich geht alles schief		396
Kapitel 39		
Die Hälfte von zehn Wegen, etwas rückgängig zu machen		397
Ich habe einige Sachen im Editor eingegeben oder gelöscht, aber eigentlich wollte ich das gar nicht		397
Ich wollte diese Option doch gar nicht ändern		397
Verdammtd, ich habe versehentlich eine Datei gespeichert, und das war falsch		398
Meine Fenster passen nicht auf den Bildschirm		398
Ich habe über den Tisch gelangt, um eine tolle CD einzulegen, und dabei eine Dose Cola über das Festplattenlaufwerk gekippt		398
Kapitel 40		
Es muss doch zehn Mittel gegen Programmabstürze geben		399
Sie haben vergessen, eine Variable zu initialisieren		399
Sie haben = anstelle von == verwendet		400
Sie haben einen Nullzeiger verwendet		400
Gute Zeiger, schlechte Zeiger		401
Sie haben vergessen, einen Zeiger zu dereferenzieren		402
Sie haben vergessen, Speicher zu belegen		402
Sie überschreiben bereits belegten Speicher		403
Sie haben eine Warnung über Konflikte bei der Konvertierung ignoriert		403
Sie haben das break in einer switch-Anweisung vergessen		404
Sie haben durch Null dividiert		405

Kapitel 41	
Zehn weitere Mittel gegen den Absturz danach	407
Sie haben vergessen, einen Schleifenzähler zu inkrementieren	407
Sie haben die Zählervariable in einer Schleife verändert	408
Falsche Abbruchbedingungen in einer Schleife	409
Sie haben die falsche Variable benutzt	409
Falsche Grenzen eines Feldes	411
[x,y] ist nicht [x][y]	412
Ändern von Variablen außerhalb des Gültigkeitsbereichs	412
oder Ändern globaler Variablen	412
Sie haben Windows-Angelegenheiten in einem CWnd-Konstruktor erledigt	413
Sie haben einen falschen Parameter an Windows übergeben	414
Eine Datei hatte eine ungültige Uhrzeitangabe oder ein ungültiges Datum	414
Kapitel 42	
Die Top Ten der MFC-Klassen	415
CWinApp: Mach doch mal einer das Fenster auf	416
CWnd: Verschieben wir's einfach	416
CDialog: Ein Schwätzchen durchs Fenster	416
CMDIFrameWnd: Immer Fenster druff	416
CToolBar: Der Werkzeugkasten	417
CDialogBar: Wenn Ihnen Schaltflächen nicht reichen	417
CStatusBar: Status quo vadis	417
CDC: Schreib' mal wieder	417
CPen: Fensterschmierereien	417
CBitmap: Andy Warhol Does Windows	417
Kapitel 43	
Die Top Ten der MFC-Elementfunktionen	419
Und hier sind sie	419
Anhang A	
Über die CD	421
Systemanforderungen	421
Wie man die CD unter Microsoft Windows benutzt	421
Wie man mit der CD umgeht	422
Was Sie finden werden	422
Visual C++ 6	422

Beispielcode	423
Werkzeuge	423
Bibliotheken	424
Wenn Sie (CD-)Probleme haben	425

Anhang B

Glossar

427

Stichwortverzeichnis

441