

INHALTSVERZEICHNIS

Vorwort	23
----------------------	-----------

Kapitel 1: Wir richten uns ein ...

Entwicklungsumgebungen für C++

Seite 25

Brauche ich eine IDE zum Programmieren?	26	Lass uns endlich loslegen	31
Multikulturelle Sachen	27	Übersetzen mit einer Entwicklungsumgebung	31
Mikroweiche Sachen	28	g++ und clang++	35
Die X-Sachen	29	... am Ende läuft es	37
Angebissene Äpfel	30		

Kapitel 2: Elefanten können nicht fliegen, aber Schrödinger kann programmieren

Erste Schritte in C++

Seite 39

Was ist eigentlich ein Computerprogramm?

Ganz kurz und knapp für alle Schrödingers	40	Auf dem Weg zum Bildschirm	50
Die Sache mit dem main-Dings	41	Badewanne schon voll?	50
Unser erstes main-Dings soll laufen	43	Gib mir fünf	51
Endlich entspannen undträumen!	46	Stream me up, Scotty	51
Kreuz und quer oder alles in Reih und Glied?	47	Gegenseitige Wahrnehmung	52
Keinen Kommentar?	47	Manipulieren oder selber steuern?	53
Wie komme ich hier zum Bildschirm ...?	48	Noch ein wenig Brain-Streaming	54

Kapitel 3: Verschiedene Typen für einen bestimmten Zweck

Die C++-Basisdatentypen

Seite 57

Starke Typen	58	Das Pünktchen in der Werkstatt	68
Mein Name ist Schrödinger	59	Am Ende war das Pünktchen	69
Fundamentale und faule Typen	59	Zeichensalat	70
Deklaration und Definition	60	Doch ein ganzer Kerl?	71
Ganzer Kerl dank	61	Turmbau zu Babel	72
Zeichenfolgen von Ganzzahlen	62	Zum Flüchten	73
Positive und/oder negative Haltung und ein Endteil	62	Unicode-Unterstützung	73
Die Sache mit der Wahrheit	63	Zeichen für die Welt	74
Was nehmen wir für einen Typen?	65	Erste „Zeichen“ für eine Pause	76
Die Welt der ganzen Kerle	66	Auf die Größe kommt es an	78
Was für den einen das Komma, ist für den anderen der Punkt	67	Je größer, desto besser	79
		The Final Frontier	81
		Gut, dass es Grenzen gibt	83

Kapitel 4: Von Zahlen verweht ...

Arbeiten mit Zahlen

Seite 85

Manueller Rechenschieber	86	Keine Ahnung, aber ich verwende es trotzdem	101
Erweiterter Grundrechenzuweisungsoperator	87	Am Ende der Mathewelt	104
Futter für den Prozessor	88	Den Typ mag ich nicht	106
Kopfrechnen	90	Lass ihn doch	106
Achtung vor den Doppelgängern	90	Automatische Promotion	107
Nachsitzen in Mathe	92	Mit dem Hammer auf die Schraube	108
Wenn Grenzen überschritten werden	92	Warum man sich nicht auf JEDEN Typ einlassen sollte	110
Ungenaues Pünktchen	94	Der sanfte Typ	112
Schwächen offenlegen	95	Automatische Typenableitung	114
Mir raucht der Kopf	97		
Mathematische Spezialitäten	99		

Kapitel 5: Eigene Entscheidungen treffen oder das Ganze nochmal bitte

Kontrollstrukturen in C++

Seite 115

Endlich selbstständig sein und eigene Entscheidungen treffen	116	Immer diese Wiederholungen	134
Verzweigung, Abzweigung oder Kreuzung	119	Ein Schritt vor oder einer zurück	134
Wrong turn?	121	After a while	135
Ein ganz anderer Fall	125	Now for it!	136
Den Fall bearbeiten	127	Fußnoten nicht vergessen!	137
Den Fall analysieren	129	Nach oben oder nach unten	137
Also sprach Zarathustra	131	Und alles noch einmal	138

Kapitel 6: Von gleichen und unterschiedlichen Typen, dem Sternchen und anderen ungemütlichen Sachen

Arrays, Strings, Vektoren, Strukturen und Zeiger

Seite 141

Gleiche Typen in einer Reihe aufstellen	142	Endlich echte Schuhdaten	159
Die Frickel- und Fummelfraktion von Ze	143	Die gemischten Typen sind echt nützlich	162
Die Ze-Strings	144	Von Unionen, Aufzählungen und Synonymen	164
Gib mir rohen Input (Bio-Arrays)	145	1, 2, Freddy kommt vorbei, 3, 4, schließe deine Tür	165
Das macht keinen Spaß	147	Die Lehre der Synonymie	169
Krankheiten von Ze-Arrays und Ze-Strings	148	Leipziger Allerlei	170
Die gleichen Typen in einer Reihe aufstellen und wieder zurück	150	typedef Schrödinger hat_alles_kapiert_t	172
Die Komfortklasse(n) von Ze++	150	Weißt du, wie viele Sternlein am Himmel stehen?...	174
Reduzierter Bioanteil in vector	151	Ich weiß, wo du wohnst ...!	175
Reduzierter Bioanteil in string	152	Einbruch in fremde Wohnungen	176
Nie mehr Bio(-Arrays)	153	Wohnorte ermitteln	176
Am Ende hat es doch noch Spaß gemacht	155	Sternenkunde	177
Die Mischlinge	157	Ze-Zeugs, Zeiger und wo soll das hinführen ...?!	178
Zugriff auf die Mischlinge	158	Zeiger auf nichts ...!	180

Wo geht's hier zur „Milky Way“?	181	Speicherhalde und Müllbeseitigung	184
Wo gibt's hier frischen RAM?	183	RAM mit WoW-Freunden auf Anfrage	186
Alles neu	183	RAM Unleashed	189

Kapitel 7: Funktionen, das Ende von Copy & Paste ...

Funktionen

Seite 193

Die Geister, die ich rufen will	194	Dinge, die man besser nicht beschwören sollte	212
Meine Hausgeister	195	Referenzen als Rückgabewert	214
Erster Kontakt zum Jenseits	197	Die Stille ist zerrissen	215
Geisterjäger im Wohnzimmer	199	Hausgeister zurückgeben	217
Opfergaben für die Geister	201	Jetzt bist du ein Medium	218
Als Kopie übergeben (Call-by-Value)	201	Spezielle Geister	220
Als Adresse übergeben (Call-by-Reference)	202	Werte für den Notfall	220
Referenzen als Funktionsparameter	203	Gleicher Name, unterschiedliche Typen	221
Ze-Zeugs als Funktionsparameter	204	Rollerblades für Funktionen?	221
Strukturen und Klassen als Parameter	205	main Programmende	223
Unsere Gaben wurden angenommen	206	Jenseits von Eden	224
Unendliche Stille	208	Am Ende der Geisterwelt	227
Das Jenseits antwortet dir	211		
Zeiger als Rückgabewert	212		

Kapitel 8: Wie aus Chaos Ordnung entsteht

Schlüsselwörter für Typen, Namensbereiche und die Präprozessor-Direktiven

Seite 229

Eigenschaften ohne Ende	230	Extrawurst-Gültigkeitsbereich	241
Klasse, die Speicherklasse	230	Einen neuen Lebensraum schaffen	242
Typqualifikationen	232	Betreten des neuen Lebensraumes	243
... und für die Funktionen auch noch		Using me	244
Extrawürstchen	233	Ein eigenes kleines Königreich	246
Mindesthaltbarkeitsdatum ändern	234	Anti-Globalisierung	249
Gleich-Gültigkeitsbereich	238	Anonymer Lebensraum	250

Lebensraum im Wohnzimmer	252	Die Zutaten für den leckeren Kuchen	266
Das #Ding vor dem Compiler	255	„Symbol(s) not found“, oder eine Zutat fehlt	267
#include „paste ohne copy“	256	Die Einkaufsliste mit den Zutaten	268
#define „Symbol“ und „Makro“	256	Ein nützliche Einkaufsliste, was alles so	
Die Übersetzung dirigieren	258	in einer Headerdatei verwendet werden	
#Ich bestimme, was #du bekommst	259	könnte/sollte/muss	268
„No such file or directory“, oder		Die Zutaten vorbereiten und abwiegen	269
wo bin ich hier	260	... und jetzt alles in die Schüssel	270
Makros und Symbole, oder doch lieber nicht?	261	Rein ins Vergnügen	271
#Ich h### all## v##sch##t	263	Meister der Quelldateien	275
Und jetzt alle zusammen!	265		

Kapitel 9: Von Hexenmeistern, Todesrittern und Datenkapseln

Klassen

Seite 277

Oben-ohne-Programmierung	278	Aufbauen und Vernichten	302
Klasse, Klassen!	279	Dienst nach Vorschrift	302
Objekte zum Mitnehmen	279	Wir übernehmen selbst	303
Ein Objekt erblickt das Licht der Welt	280	Konstruktoren mit mehreren Parametern	304
Kontrolle: Du kommst hier nicht durch	282	Konstruktoren effektiver initialisieren	305
Bei Klassendefinitionen den		Klassenelemente direkt initialisieren	306
Überblick behalten	284	Am Ende alles wieder saubermachen	306
Tieferer Einblick in die Elementfunktionen der		Frühjahrsputz	307
Datenkapsel	286	(K)ein Kartenhaus	310
Du darfst hier nur lesen	288	Deep inside	312
Elementfunktionen voll im Einsatz	289	Spezielle Konstruktoren	312
Toll, diese Klassenfunktionen	291	Praxis Dr. Schrödinger	315
Objekte erstellen	293	Wohnung von Dr. Schrödinger	317
Objekte auf die Welt bringen	294	The Big Three	318
Zugriff auf die Öffentlichkeit der Klasse	295	Spezielle Daten in der Kapsel	319
Indirekter Zugriff auf die Öffentlichkeit	295	Gute Freunde kann niemand trennen	321
Objekte verwenden	296	*Gong* Die letzte Runde wird eingeläutet	323
Die Geschichte von Objekten	299	Kampfanalyse	328

Kapitel 10: Kino + WoW + Programmieren = viel Spaß

Überladen von Operatoren

Seite 331

Eigene Rechengesetze	332	Weitere Operatoren überladen	351
Gestatten: operator	332	Logisch? Fast immer!	351
Gesetze für die Herren Operatoren	333	„Typenverbiegenumwandler“ überladen	351
Operatoren nur für Freunde	334	Input-Output-Kompott	352
Die Pärchen verwenden die Operatoren	335	Spezielle Operatorüberladungen in der Praxis	354
Mit vereinten Kräften	337	Spezialitäten auf dem Sofa	360
Glückliche Pärchen	340	Funktionsobjekte	360
Einsame Operatoren überladen	344	Indexoperator [] überladen	361
Das einsame Leben der einsamen Operatoren	346	new und delete überladen	361
Am Ende bleibt ein einsamer Operator	349		

Kapitel 11: Schrödinger macht sein Testament

Abgeleitete Klassen

Seite 363

Erben ohne Erbschaftssteuer	364	Wer bekommt was	392
Ewig schleichen die Erben	367	Keiner geht leer aus	394
Damit keiner leer ausgeht	371	Mehrfachvererbung	396
Jetzt das Kleingedruckte lesen	373	Mehrfachvererbung in der Praxis	398
Zugriffsrechte für den Beerbtan	373	Lohnt sich die Mehrfachvererbung überhaupt? ..	401
News: Konstruktoren vererben (C++11)	376	Virtuelles Vererben	403
Das Kleingedruckte in der Praxis	377	Virtuelle Teilchen verwenden	405
So macht erben langsam Spaß	380	Zwischen Virtualität und Realität	406
Private Mitglieder durchreichen	381	Abstrakte Welten	408
Erbe verwenden und erweitern	382	Ein alter Bekannter	408
Redefinition	383	Abstrakte Vielgestaltigkeit	410
Use me	383	Jetzt wird es vielseitig – Polymorphie	411
Unser Anlageberater verwaltet das Erbe	385	Virtuelle Zerstörungskraft	413
Ordentlich angelegt	387	Was vom Abstrakten übrig blieb	414
Konstruktives und destruktives Vererben	389	Was bist du denn?	415
Implizite Klassenumwandlung	391	override und final	416

Kapitel 12: Ausstechformen für die Plätzchen

Templates

Seite 419

Funktionen zum Ausstechen	420	Klassen-Ausstecher-Elementfunktion definieren ..	430
Verschiedene Teigsorten	422	Klassen-Ausstecher-Elementfunktion überschreiben ..	431
Plätzchen backen	424	Objekte ausstechen	432
Am Ende sind nur noch Krümel da	428	Klassen-Ausstecher in der Praxis	433
Klassen zum Ausstechen	429	Klassen-Ausstecher in der Wohnung	438

Kapitel 13: Der Schleudersitz für den Notfall

Ausnahmebehandlung

Seite 441

Versuchen, werfen und auffangen	442	Wir probieren es aus	464
Noch ein paar Hinweise für das Werfen	444	Logischer Fehler: out_of_range	464
Jetzt schmeiß schon!	446	Logischer Fehler: invalid_argument	465
Was passiert danach ...?	447	Logischer Fehler: length_error	466
Homerun	450	Logischer Fehler: ios_base::failure	467
Mit Klassen um sich schmeißen	453	Standardausnahme-Kontrolle	468
Nix wie weg hier	453	Ausnahme-Spezifikation und noexcept?	470
Schmeißen mit ganzen Klassen	455	noexcept	470
Homerun mit Klassen	458	Hasta la vista, baby	471
(Standard-)Ausnahmen im Angebot	461	Ausnahmen verweigern	472
What ist dein Problem ...?	462	Keine Erwartungen	474
Ausnahmen im System	463		

Kapitel 14: Unterwäsche, 100 % Baumwolle, Doppelripp

Die Standardklasse `string`

Seite 475

Schurz, Schürzen, Schürzenjäger	476	Wie groß isses denn?	480
Strings anlegen und zuweisen	476	Rohe Strings	480
Zugriff auf die einzelnen Zeichen	479	Noch mehr Unterwäsche	481

Und noch mehr davon	483	Überladene Operatoren und Ein-/Ausgabe	488
Klamottenkiste	485	Ich kann's nicht mehr hören: Strings	490
String konvertieren und manipulieren	485	Alles sauber dank „Schwarzer Zwerg“	494
Such!	486		

Kapitel 15: Ströme ohne Isolierkabel verwenden

Der Umgang mit Streams und Dateien

Seite 495

Gib aus den Strom	496	Datei-Slang	526
Rohe Eier raus ...	498	Vorhandene Ströme für Dateien	527
Mehr formatierter Output bitte	499	Strom für die Datei anschließen	527
Wir schwenken die Flagge ...	500	Plug-ins für den Dateistrom	528
Jetzt ist es raus ...	502	Den Dateistecker ziehen	530
Wir ändern die Richtung	505	Alles gesichert ...?	531
Rohe Eier rein ...	505	Sauber lesbare Sache	531
Wo ist mein Input?	507	Stück für Stück	532
Jetzt ist es drin ...	511	Zeile für Zeile	534
Wir manipulieren die Ströme	513	Ganze Happen ...	535
Manipulation ist alles ...	516	Wahlfreier Zugriff	536
Ordentlich manipuliert ...	520	Daten wiederherstellen	537
Auch ein Strom hat seine Fehler	522	Ein Strom für einen String?	541
Erst den Strom abstellen ...	524	Schürzenjäger-Strom	543
Die Sicherung ist geflogen	525	Ohne Isolation	545
Kann man auch was speichern?	526		

Kapitel 16: Aussteckformen für Faule

Einführung in die Standard Template Library (STL)

Seite 547

Fertigkuchen von Dr. STL ...?	548	Detaillierteres Arbeiten mit sequenziellen	
Verschiedene Behälter (Container)	549	Fertigkuchen	558
Algorithmen und Iteratoren	551	Behälter erstellen	558
Besser als „Selbermachen“	552	Zutaten hinzufügen	559
... und schmeckt auch noch gut!	555	Zugriff auf den Teig	560

Wie viel passt rein, und wie viel ist drin ...?	561	Funktionsobjekte	581
Raus damit ...!	561	Kategorie von Iteratoren	583
Tausch mit mir, oder gib mir alle	562	Iterator-Adapter	585
Mixen, sortieren und rühren	562	Die Hilfsmittel für Fertigkuchen und Zutaten im Einsatz	586
Sequenzielle Fertigkuchen abschmecken	563	Hilfe für den Iterator	588
Bereit zum Essen	568	Allmählich wird es öde	589
Detaillierteres Arbeiten mit assoziativen Fertigkuchen	570	Die fleißigen Arbeiter	592
set und multiset	571	Nicht-modifizierende Algorithmen	592
map und multimap	572	Modifizierende Algorithmen	593
Bitte ein Bit-Feld ...!	572	Löschende Algorithmen	594
Assoziative Kuchen backen	573	Mutierende Algorithmen	595
(multi)set me up, baby!	573	Sortierende Algorithmen	596
Now (multi)map me!	575	Algorithmen für sortierte Bereiche	597
Bitte ein Bit!	576	Algorithmen verwenden	598
Auch assoziative Kuchen kann man essen	579	Ende gut, alles gut	602
Zwischen Fertigkuchen und weiteren Zutaten	581		

Kapitel 17: Schöne neue Welt C++11

C++11 – der neue Standard

Seite 605

C++ auf dem neuesten Stand(ard)	606	Cool, das neue Zeugs	616
auto/decltype	607	Weitere nützliche Features	617
Einfachere Initialisierung	607	Noch mehr Neuigkeiten	619
Lambda-Funktionen	608	Ein neues Array?	619
Range-based-loop	608	Eine neue verkettete Liste?	620
Explizite delete- und default-Funktionen	609	Hasch? Ist das nicht illegal?!	620
nullptr	609	Neue Algorithmen	621
constexpr	610	Tuple? Tulpe?	622
Ein Konstruktor ruft einen anderen Konstruktor auf	611	Neue Planeten braucht das Universum	623
Move your body	611	Neue Backmischungen sind auch gut	626
Neues Zeugs im Einsatz	613	Kluge Zeiger	628
auto/decltype	613	Ich bin der Klügste hier (shared_ptr)	630
{}-Initialisierer verwenden	613	Schwacher Zeiger (weak_ptr)	631
Lambda-Funktion	614	Egoistischer, aber kluger Zeiger (unique_ptr)	633
Move my own class	615	Klug auch in der Praxis	634
		Bist du auch so klug ...?	637

Von Hieroglyphen und regulären Ausdrücken	639
Mini-Sprachkurs zu den Hieroglyphen	639
Objekt für die Hieroglyphen	643
Die Algorithmen für Hieroglyphen	643
Suchergebnis analysieren	644
Suchen mit Hieroglyphen	645
Cleopatra ist da	652
Parallele Welten	654
Viele Fäden erzeugen	655
Bist du jetzt ein Faden oder nicht?	657
Argumente für den Thread	657

Wir nehmen jetzt die Fäden in die Hand	658
Nur nicht den Faden verlieren	661
Schütze deine Daten	663
Ein Schloss für den Mutex	666
Sicheres Initialisieren	667
Totgesperrt	668
Einmal bitte	670
Am Ende des Fadens	672
„Konditions“-Variable ...?	672
Zusammenfassung	674

Kapitel 18: C++ 14 – der Neue!

C++14 – der allerneueste Standard

Seite 675

Schon wieder ein neuer Standard?	676
Der Compiler weiß es doch sowieso immer besser	677
Dann mach es doch auch selbst bei den λ -Funktionen	677
Gammelcode an den Compiler verraten?	678
Etwas für die Bitverdrehner unter uns	679

Und dann noch etwas für die Zahlenzerstückler	680
Mr. Holmes, bitte übernehmen Sie	681
Der Tanz mit den Lambda-Parametern	682
Alte Sachen aufheben oder ausmisten?	683
Mir reicht es jetzt mit der neuen Welt	684
Noch ein paar mehr C++14-Sachen	685

Index	689
--------------------	------------