

Vorwort

Die 5. Auflage unterscheidet sich von der 4. Auflage durch die Umstellung auf den 2017 von der zuständigen ISO/IEC-Arbeitsgruppe JTC1/SC22/WG21 verabschiedeten C++-Standard. Die Veränderungen gegenüber dem C++-Standard von 2014 sind durch Markierungen am Seitenrand (wie hier) gekennzeichnet. Das Buch ist konform zum neuen C++-Standard, ohne den Anspruch auf Vollständigkeit zu erheben – das Standard-Dokument [ISO C++] umfasst allein mehr als 1600 Seiten. Sie finden in diesem Buch eine verständliche und mit vielen Beispielen angereicherte Einführung in die Sprache. Im Teil »Das C++-Rezeptbuch« gibt es zahlreiche Tipps und Lösungen für typische Aufgaben, die in der täglichen Praxis anfallen. Es gibt konkrete, sofort umsetzbare Lösungsvorschläge. Zahlreiche Algorithmen für praxisnahe Problemstellungen helfen bei der täglichen Arbeit. Auf größtmögliche Portabilität wird geachtet: Die Beispiele funktionieren unter Linux genauso wie unter Windows und OS X. Die problembezogene Orientierung lässt die in die Sprache einführenden Teile kürzer werden. Damit wird das Lernen erleichtert und die Qualität des Buchs, auch als Nachschlagewerk dienen zu können, bleibt erhalten.

C++17

Für wen ist dieses Buch geschrieben?

Dieses Buch ist für alle geschrieben, die einen kompakten und gleichzeitig gründlichen Einstieg in die Konzepte und Programmierung mit C++ suchen. Es ist für Anfänger¹ gedacht, die noch keine Programmiererfahrung haben, aber auch für andere, die diese Programmiersprache kennenlernen möchten. Beiden Gruppen und auch C++-Erfahrenen dient das Buch als ausführliches Nachschlagewerk.

Die ersten zehn Kapitel führen in die Sprache ein. Es wird sehr schnell ein Verständnis des objektorientierten Ansatzes entwickelt. Die sofortige praktische Umsetzung des Gelernten steht im Vordergrund. C++ wird als Programmiersprache unabhängig von speziellen Produkten beschrieben. C-Kenntnisse werden nicht vorausgesetzt. Das Buch eignet sich zum Selbststudium und als Begleitbuch zu einer Vorlesung oder zu Kursen. Die vielen Beispiele sind leicht nachzuvollziehen und praxisnah umsetzbar. Klassen und Objekte,

¹ Geschlechtsbezogene Formen meinen hier und im Folgenden stets Männer, Frauen und andere.

Templates und Exceptions sind Ihnen bald keine Fremdworte mehr. Es gibt mehr als 90 Übungsaufgaben – mit Musterlösungen im Anhang. Durch das Studium dieser Kapitel werden aus Anfängern bald Fortgeschrittene.

Diesen und anderen Fortgeschrittenen und Profis bietet das Buch kurze Einführungen in die Themen Thread-Programmierung, Netzwerk-Programmierung mit Sockets einschließlich eines kleinen Webservers, Datenbank-Anbindung, grafische Benutzeroberflächen und mehr. Dabei wird durch Einsatz der Boost-Library und des Qt-Frameworks größtmögliche Portabilität erreicht.

Softwareentwicklung ist nicht nur Programmierung: Einführend werden anhand von Beispielen unter anderem die Automatisierung der Programmierzeugung mit Make, die Dokumentationserstellung mit Doxygen und die Versionskontrolle behandelt. Das Programmdesign wird durch konkrete Umsetzungen von UML²-Mustern nach C++ unterstützt. Das integrierte »C++-Rezeptbuch« mit mehr als 150 praktischen Lösungen, das sehr umfangreiche Register und das detaillierte Inhaltsverzeichnis machen das Buch zu einem praktischen Nachschlagewerk für alle, die sich mit der Softwareentwicklung in C++ beschäftigen.

Moderne Programmiermethodik

Beim Programmieren geht es nicht nur darum, eine Programmiersprache zu lernen. Es geht auch darum, Programme zu schreiben, die hohen Qualitätsansprüchen gerecht werden. Dazu gehört das Know-how, die Mittel der Sprache richtig einzusetzen. Dass ein Programm läuft, reicht nicht. Es soll auch gut entworfen sein, möglichst wenige Fehler enthalten, selbst mit Fehlern in Daten umgehen können, verständlich geschrieben und schnell in der Ausführung sein. In diesem Sinn liegt ein Schwerpunkt des Buchs auf der Methodik des Programmierens, unterstützt durch Darstellung der informatischen Grundlagen, viele Hinweise im Text und demonstriert an vielen Beispielen.

Übersicht

Schwerpunkt von Teil I ist die Einführung in die Programmiersprache. Die anschließenden Teile gehen darüber hinaus und konzentrieren sich auf die verschiedenen Probleme der täglichen Praxis. Die in dieser Übersicht verwendeten Begriffe mögen Ihnen zum Teil noch unbekannt sein – in den betreffenden Kapiteln werden sie ausführlich erläutert.

Teil I – Einführung in C++

Kapitel 1 vermittelt zunächst die Grundlagen, wie ein Programm geschrieben und zum Laufen gebracht wird. Es folgen einfache Datentypen und Anweisungen zur Kontrolle des Programmablaufs. Die Einführung der C++-Datentypen `vector` und `string` und einfache Ein- und Ausgaben beenden das Kapitel.

Kapitel 2 zeigt Ihnen, wie Sie Funktionen schreiben. Makros, Templates für Funktionen und die modulare Gestaltung von Programmen folgen.

Objektorientierung ist der Schwerpunkt von *Kapitel 3*. Dabei geht es nicht nur um die Konstruktion von Objekten, sondern auch um den Weg von der Problemstellung zu Klassen und Objekten. Zeiger sowie die Erzeugung von Objekten zur Laufzeit sind Inhalt von

² UML ist eine grafische Beschreibungssprache für die objektorientierte Programmierung.

Kapitel 4. Kapitel 5 führt das Thema Objektorientierung fort. Dabei erfahren Sie, wie eine String-Klasse funktioniert, und wie Sie Klassen-Templates konstruieren. *Kapitel 6* zeigt Ihnen das Mittel objektorientierter Sprachen, um Generalisierungs- und Spezialisierungsbeziehungen auszudrücken: die Vererbung mit ihren Möglichkeiten. Strategien zur Fehlerbehandlung mit Exceptions finden Sie in *Kapitel 7. Kapitel 8* zeigt, wie Sie Operatorsymbolen wie + und - eigene Bedeutungen zuweisen können und in welchem Zusammenhang das sinnvoll ist. Sie lernen, »intelligente« Zeiger (Smart Pointer) zu konstruieren und Objekte als Funktionen einzusetzen. *Kapitel 9* beschreibt ausführlich die Ein- und Ausgabemöglichkeiten, die vorher nur einführend gestreift wurden, einschließlich der Fehlerbehandlung und der Formatierung der Ausgabe. Eine Einführung in die Standard Template Library (STL) bietet *Kapitel 10*. Die STL und ihre Wirkungsweise bilden die Grundlage eines sehr großen Teils der C++-Standardbibliothek.

Teil II – Fortgeschrittene Themen

In diesem Teil folgen fortgeschrittene Themen wie Wert- und Referenzsemantik (*Kapitel 11*), Lambda-Funktionen (*Kapitel 12*), Template-Metaprogrammierung (*Kapitel 13*), reguläre Ausdrücke (*Kapitel 14*) und die Programmierung paralleler Abläufe mit Threads (*Kapitel 15*). *Kapitel 16* zeigt, wie grafische Benutzungsschnittstellen konstruiert werden. Wie ein Programm die Verbindung mit dem Internet aufnehmen kann, dokumentiert *Kapitel 17*. Und wohin mit den ganzen Daten, die bei Programmende nicht verloren gehen sollen? In *Kapitel 18* lernen Sie, wie ein Programm an eine Datenbank angebunden wird. Manche der genannten Themen sind so umfangreich, dass sie selbst Bücher füllen. Für diese Themen wird daher nur ein Einstieg geboten.

Teil III – Ausgewählte Methoden und Werkzeuge der Softwareentwicklung

Die Entwicklung von Programmen besteht nicht nur aus dem Schreiben von Code. Die Compilation eines Projekts mit vielen Programmdateien und Abhängigkeiten kann schnell ein komplexer Vorgang werden. Die Automatisierung dieses Prozesses mit dem Tool *make* ist Thema von *Kapitel 19*. Programme sind nicht auf Anhieb fehlerfrei. Sie müssen getestet werden: *Kapitel 20* stellt ein Werkzeug für den Unit-Test vor und zeigt den praktischen Einsatz.

Teil IV – Das C++-Rezeptbuch: Tipps und Lösungen für typische Aufgaben

Sichere Programmentwicklung ist die Überschrift von *Kapitel 21*. Sie finden dort Regeln zum Design von Methoden und Tipps zur defensiven Programmierung, die die Wahrscheinlichkeit von Fehlern vermindern. *Kapitel 22* zeigt Rezepte, wie Sie bestimmte UML-Muster in C++-Konstruktionen umwandeln können. Algorithmen für viele verschiedene Aufgaben finden Sie in *Kapitel 23*. Wegen der Vielzahl empfiehlt sich ein Blick in das Inhaltsverzeichnis, um einen Überblick zu gewinnen. *Kapitel 24* enthält fertige Rezepte zum Anlegen, Löschen und Lesen von Verzeichnissen und mehr.

Teil V – Die C++-Standardbibliothek

Hier wird die C++-Standardbibliothek in Kürze beschrieben. Die Inhalte dieses Teils sind: Hilfsfunktionen und -klassen, Container, Iteratoren, Algorithmen, Einstellung nationaler Besonderheiten, String, Speicherverwaltung, Funktionen der Programmiersprache C.

Anhang

Der Anhang enthält unter anderem verschiedene hilfreiche Tabellen und die Lösungen der Übungsaufgaben.

■ Wo finden Sie was?

Bei der Programmentwicklung wird häufig das Problem auftauchen, etwas nachschlagen zu müssen. Es gibt die folgenden Hilfen:

Erklärungen zu Begriffen sind im *Glossar* ab Seite 963 aufgeführt.

Es gibt ein recht umfangreiches *Stichwortverzeichnis* ab Seite 977 und ein sehr detailliertes *Inhaltsverzeichnis*.



Software zum Buch

Auf der Webseite <http://www.cppbuch.de/> finden Sie die Software zu diesem Buch. Sie enthält unter anderem einen Compiler, eine Integrierte Entwicklungsumgebung sowie alle Programmbeispiele und die Lösungen zu den Aufgaben. Sie finden dort auch weitere Hinweise, Errata und nützliche Links.

■ Zu guter Letzt

Allen Menschen, die dieses Buch durch Hinweise und Anregungen verbessern halfen, sei an dieser Stelle herzlich gedankt. Frau Brigitte Bauer-Schiewek und Frau Irene Weilhart vom Hanser Verlag danke ich für die gute Zusammenarbeit.

Bremen, im Oktober 2017

Ulrich Breymann