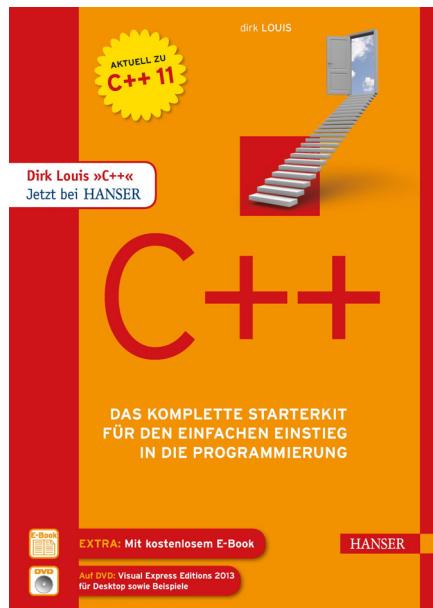


# HANSER



## Stichwortverzeichnis

zu

„C++“

von Dirk Louis

ISBN (Buch): 978-3-446-44069-2

ISBN (E-Book): 978-3-446-44109-5

Weitere Informationen und Bestellungen unter  
<http://www.hanser-fachbuch.de/978-3-446-44069-2>  
sowie im Buchhandel

# Index

## Symbolen

\ 39, 191  
- \  
  - (Gültigkeitsbereichsoperator)  
  16, 119  
+ (Addition) 62, 372  
+ (Addition, string) 68, 195 f.  
& (Adresse) 148  
\a (Signalton) 192  
<< (Ausgabe, cout) 203, 373  
<< (Ausgabe, ostream) 203  
<< (Ausgabe, stringstream) 201  
~ (bitweises Komplement) 357  
| (bitweises ODER) 356  
& (bitweises UND) 356  
^ (bitweises XOR) 356  
- (Dekrement) 67  
/ (Division) 62  
>> (Eingabe, cin) 210, 373  
>> (Eingabe, istream) 210  
>> (Eingabe, stringstream) 201  
#endif 256  
\ (Escape-Zeichen) 192  
Exceptions . Siehe Ausnahmen  
== (Gleichheit) 73  
== (Gleichheit, string) 197  
> (Größer) 73  
>= (Größer oder gleich) 73  
>= (Größer oder string) 197  
> (Größer, string) 197  
#ifndef 15  
#include 15  
[] (Indizierung, string) 199  
[] (Indizierung, vector) 239  
++ (Inkrement) 67, 371  
++ (Inkrement, Iteratoren) 242  
< (Kleiner) 73  
<= (Kleiner oder gleich) 73  
<= (Kleiner oder gleich, string)  
  197

< (Kleiner, string) 197  
// (Kommentar, einzeilig) 27  
/\* (Kommentar, mehrzeilig) 27  
<< (Linksverschiebung) 357  
! (logisches NICHT) 75  
|| (logisches ODER) 75  
&& (logisches UND) 74  
% (Modulo) 62  
\* (Multiplikation) 62  
\n (Zeilenumbruch) 192  
#pragma once 258, 275  
>> (Rechtsverschiebung) 357  
- (Subtraktion) 62  
\t (Tabulator) 192  
!= (Ungleichheit) 73  
!= (Ungleichheit, string) 197  
- (Vorzeichen) 62  
+ (Vorzeichen) 62

## A

Ablaufsteuerung 71  
Ableitung 305  
Adressen  
- in Zeigern speichern 147f.  
- von Variablen 148  
Algorithmen (STL) 243  
- generate() 246  
Algorithmus 11  
Anweisungen 24  
- leere Anweisungen 97  
Arbeitsspeicher 49  
argc 114  
Argumente  
- aus Befehlszeile 115  
argv 114  
Arrays 125, 236  
- als Parameter 131  
- Anzahl Elemente 126, 128  
- array-Container (C++11) 234

- Definition 126  
- Elemente  
  - durchlaufen 128  
  - Zugriff 126  
- Indizierung 126  
- Initialisierung 127  
- mehrdimensionale 131  
- sortieren 248  
- Speicherverwaltung 126  
- Typ 126  
- Variable 126  
- von Basisklassentypen 330  
- von Objekten 299  
- von Strukturen 140  
asctime() 225  
Assembler 6  
Aufzählungen 132  
- Definition 135  
- enum 132  
- enum class (C++11) 136  
- in switch-Verzweigungen 136  
Ausdrücke 64  
- boolesche 72  
- einfache 65  
- Klammern 66  
- komplexe 65  
- Nebeneffekte 98  
Ausgabe 203  
- cout 25, 203  
- Formatierung 204  
- Genauigkeit von Gleitkomma-zahlen 205  
- in Binärdateien 220  
- in Textdateien 214  
Ausnahmen  
- abfangen 342  
- auslösen 345  
- catch-Blöcke 342 f.  
- cin 213, 341  
- eigene Ausnahmeklassen 345  
- noexcept (C++11) 346

- Parameter 345
- Programmfluss 347
- stack unwinding 348
- Standardbibliothek 344
- throw 345
- try-Block 342
- weiterleiten 346
- auto (C++11) 241

## B

- Backslash-Zeichen 192
- bad() (basic\_ios) 212
- badbit (ios\_base) 211
- basic\_ios (Klasse)
  - bad() 212
  - clear() 212
  - eof() 212
  - fail() 212
  - good() 212
  - rstate() 212
- Basisklassen
  - abstrakte 335
  - -arrays 330
  - -konstruktoren 314 f.
  - -parameter 332
- Basisklassenunterobjekte 299, 310
- Initialisierung 314
- Zugriff 311
- Bedingungen 71
- Beenden
  - Funktionen 110
- Befehlszeilenargumente 115
- begin() (Container) 241
- Beispiele, auf Buch-DVD 427
- Benutzereingaben
  - einlesen (cin) 210
  - prüfen 79
- Bezeichner
  - Groß- und Kleinschreibung 44
  - Regeln für die Namensgebung 44
  - vollqualifizierte 16
- Bibliotheken
  - C++-Standardbibliothek 14
  - Pfade 179
  - verwenden 179
- Bildschirmausgaben 25
- Binärkodierung 357
- Binärsystem 50
- Binärzahlen 40
- Bindung 327
  - frühe 327
  - späte 328
- Bit 51

- Bit-Manipulation 356
  - Division 358
  - Flags 359
  - gerade Zahlen 359
  - Klein- und Großschreibung 358
  - Multiplikation 358
- Bogenmaß 186
- bool 53, 73, 436
- Boolescher Typ (bool) 73
- break 81, 93
- Buch-DVD 425
  - Beispiele 427
  - Visual Studio 425

## C

- C 7
- C++ 8
  - Compiler 18
  - Headerdateien 14
  - Programmgerüst 21
  - Standardbibliothek 14
  - Syntaxreferenz 435
- C++11
  - array-Container 234
  - auto 241, 401
  - enum class-Aufzählungen 136
  - for-Schleife 129
  - Initialisierung 46, 58, 127
    - Initialisierungslisten 238
    - Membervariablen 280
  - Lambda-Ausdrücke 401
  - long long 53
  - noexcept 346
  - override 326
  - reguläre Ausdrücke 393
  - String-Literale 194
  - Zeitnahme 224, 230
  - Zufallszahlen 189
- call by reference 154, 157
- call by value 112
- capacity() (vector) 239
- case 81
- catch 342
- char 53, 436
- cin 55, 210
  - >> 210
  - Ausnahmen 213, 341
  - clear() 212
  - Fehlerbehandlung 211
  - get() 238
  - peek() 238
- class 269
- clear() (basic\_ios) 212
- clear() (cin) 212

- clear() (string) 199
- clear() (stringstream) 201
- clock() 229
- compare() (string) 197
- Compiler
  - Compiler-Schalter 256
  - Fehlermeldungen 168
  - GNU g++ 34
  - Headerdateien 14
  - Übersetzungseinheit 256
  - Visual Studio 28
- const 48
- Memberfunktionen 292
- Membervariablen 286
- Parameter 158
- Referenzen 157
- Variablen 48
- Zeiger 157
- const\_cast 59
- Container 233
  - [] 239
  - array (C++11) 234, 451
  - deque 246
  - erzeugen 237
  - Größenmanagement 238
  - Initialisierungslisten (C++11) 238
  - Kapazität 239
  - list 241, 249
  - Memberfunktionen 240
  - vector 237
- continue 93
- cout 25, 55, 203
  - << 203
  - Fehlerbehandlung 211
  - fill() 205
  - precision() 205
  - width() 204
- c\_str() (string) 199

## D

- dangling else 99
- Dateien
  - Binärdateien 214, 220
  - kopieren 222
  - Textdateien 214
    - lesen 217
    - schreiben 214
- Datenstrukturen
  - Arrays 236
  - Hashtabellen 236, 248
  - Listen 236
  - Mengen 236
  - Stapel 236
  - Warteschlangen 236

Datentypen 42, 49  
 - Arrays 125  
 - Aufzählungen 132, 136  
 - auto (C++11) 241, 401  
 - Bedeutung 49  
 - bool 53, 73, 436  
 - char 53, 436  
 - Darstellung im Arbeitsspeicher 49  
 - double 53, 436  
 - elementare 43, 53  
 - float 53, 436  
 - int 53, 353, 436  
 - Klassen 141, 261, 268  
 - long 53, 436  
 - long long (C++11) 53, 353, 436  
 - Operandenkonvertierung 355  
 - short 53, 353, 436  
 - Strukturen 137  
 - typedef 241  
 - Typumwandlung 55  
 - Überlaufverhalten 354  
 - Übersicht 54  
 - unsigned 353  
 - Vergleiche 73  
 Datumsanzeige 223  
 Debuggen 174  
 - Visual Studio 174  
 - Vorgehensweise 174  
 default 81  
 Definition  
 - Arrays 126  
 - Aufzählungen 135  
 - auto (C++11) 241  
 - Klassen 268  
 - Memberfunktionen 288  
 - Membervariablen 277  
 - Referenzen 153  
 - Strukturen 138, 142  
 - typedef 241  
 - Variablen 42  
 - Zeiger 148  
 Deklaration  
 - Funktionen 106, 254  
 - Memberfunktionen 272  
 - Variablen 255  
 Dekrement 67  
 delete 160, 301  
 delete[] 164  
 deprecated 346  
 deque (Container-Klasse) 246  
 Dereferenzierung 150  
 Destruktoren 301  
 - Vererbung 319  
 - virtuelle 319  
 difftime() 230  
 Divide-and-Conquer-Technik 170

Division 63  
 do 88  
 double 53, 436  
 do-while 88  
 dynamic\_cast 59, 333  
 Dynamische Speicherreservierung 159  
 - Heap 159  
 - new 160  
 - new[] 164  
 - Tipps 164

**E**

Eingabe 210  
 - aus Binärdateien 220  
 - aus Textdateien 217  
 - cin 210  
 - Vorschau 238  
 Ein- und Ausgabe  
 - cin 55, 210  
 - cout 55, 203, 210  
 - Fehlerbehandlung 211  
 - printf() 206  
 Eingabeaufforderung 25, 36  
 Eingabetaste 37  
 Eintrittsfunktion main() 23, 114  
 else 78  
 empty() (Container) 240  
 end() (Container) 241  
 endl 192  
 Endlosschleifen, gewollte 95  
 Entscheidungen 71  
 enum 135  
 eof() (basic\_ios) 212  
 eofbit (ios\_base) 211  
 erase() (Container) 240  
 erase() (string) 199  
 Ersatzkonstruktor 298  
 Escape-Sequenzen 192  
 exception (Klasse) 344 f.  
 exceptions() (cin) 341  
 Exponentialschreibweise 41

**F**

fail() (basic\_ios) 212  
 failbit (ios\_base) 211  
 fail() (cin) 341  
 failure (ios\_base) 213  
 Fakultät 364  
 false 53  
 Fehlerbehandlung  
 - cin 211  
 - cout 211  
 - mit Ausnahmen 339 f.

- mit if 79  
 - Nebeneffekte 98  
 Fehlermeldungen 168  
 fill() (ostream) 205  
 find\_first\_of() (string) 199  
 find\_last\_of() (string) 199  
 find() (string) 199  
 float 53, 436  
 for 89, 129  
 friend 369  
 Function overhead 120, 289  
 Funktionen 103  
 - Arrays als Parameter 131  
 - Aufruf 104  
 - aufrufen 111  
 - beenden 110  
 - call by reference 154  
 - call by value 112  
 - Definition 104  
 - Deklaration 106, 254  
 - Function overhead 120  
 - generische 388  
 - main() 23, 114  
 - mathematische 64, 183  
 - Modularisierung 254  
 - noexcept (C++11) 346  
 - Ort der Definition 105  
 - Parameter 111  
 - Rekursion 363  
 - return 96, 109  
 - Rückgabewert 109  
 - Signatur 121, 361  
 - throw 347  
 - trigonometrische 186  
 - über Dateigrenzen hinweg verwenden 254  
 - Überladung 121  
 - Überschreibung 318, 326  
 - Vorgabeargumente 113  
 - Zeiger auf 361

## G

Ganzzahlen  
 - Division 63  
 - Literale 40  
 - Modulo 63  
 - Speicherverwaltung 50  
 generate() (Algorithmen) 246  
 Geschichte der Programmiersprachen 4  
 get() (cin) 238  
 get() (ifstream) 219  
 getline() (ifstream) 219  
 Gleitkommazahlen  
 - Division 63

- Exponentialschreibweise 41
- Genauigkeit 205
- Literale 41
- Modulo 63
- Speicherverwaltung 51
- gmtm() 224
- good() (basic\_ios) 212
- goodbit (ios\_base) 211
- goto 96
- Greenwich Mean Time 224
- Groß- und Kleinschreibung 44
- Gültigkeitsbereiche 118
  - Block 383
  - Datei 116, 383
  - Funktion 116, 383
  - Klasse 383
  - Namensraum 384
  - Sichtbarkeit 119
  - Verdeckung 119, 384

## H

- Hashtabellen 236, 248
- Headerdateien 14
  - der Standardbibliothek 15
  - einkopieren 15
  - für Klassen 272
  - #include 15
  - Mehrfachaufrufe verhindern 256
- Heap 159
- Hexadezimalzahlen 40
- Hilfe 168
  - bei Laufzeitfehlern 173
  - bei Problemen 170
  - Compiler-Meldungen 168
  - Fehlermeldungen 168
  - zu Bibliothekslementen 170
- Hollerith, Hermann 4

## I

- if 76
- if-Anweisung 76, 80
- if-else-Anweisung 78
- Initialisierung
  - Arrays 127
  - Basisklassenunterobjekte 314
  - C++11 46, 58, 127, 280
  - Membervariablen 278
  - Membervariablen (C++11) 280, 326
  - Referenzen 153
  - Strukturen 140
  - Variablen 46
  - Zeiger 148

- Inkrement 67
- inline 276, 289
- insert() (Container) 240f.
- insert() (string) 199
- Instanzen *siehe* Objekte
- Instanzierung 266, 297
- Instanzvariablen 278
- int 53, 353, 436
- Integer 40
- Integral Promotion 355
- ios\_base (Klasse) 211
  - badbit 211
  - eofbit 211
  - failbit 211
  - failure 213
  - goodbit 211
- ISO-Standard 18
- istream (Klasse) 210
  - >> 210
- Iteratoren 240

## K

- KDE 25
- Klammern
  - in Ausdrücken 66
  - runde 66
- Klassen 141, 261
  - abgeleitete 305
  - abstrakte 335
  - abstrakte Basisklassen 335
  - Basisklassen 305
  - Basisklassenunterobjekte 310
  - Definition 268
  - Destruktoren 301, 319
  - eingebettete Objekte 284
  - erlaubte Elemente 268
  - Friends 369
  - get/set-Memberfunktionen 293
  - Gültigkeitsbereich 383
  - Headerdatei 272
  - Instanzierung 266, 297
  - Instanzvariablen 278
  - Klassenvariablen 255, 287
  - Komposition 309
  - Konstruktoren 142, 267, 296
  - Kopierkonstruktur 378, 380
  - Memberfunktionen 142, 288
  - Membervariablen 142, 277
  - Modularisierung 254
  - Objekte 265
  - öffentliche Schnittstelle 271, 284
  - Polymorphie 323
  - Quelltextdatei 273

- Templates 389
- this-Zeiger 290
- über Dateigrenzen hinweg verwenden 254
- Verdeckung 318
- Vererbung 309
- Zeiger auf Objekte 268, 297
- Zugriffsspezifizierer 269
- Zuweisungsoperator 378, 380
- Klassenvariablen 255, 287
- Kommentare 27
- Kompilierung
  - Linker 18
  - Vorkompilierte Header 30
- Komplexe Zahlen 189
- Komposition 308
- Konkatenation 69, 196
- Konssole 25
- Konsolenanwendungen
  - Ausgabe (cout) 203
  - Eingabe (cin) 210
  - Menüs 82
- Konstanten 39, 286
  - const-Variablen 48
  - Literale 39
  - Strings 191
- Konstruktoren 142, 267, 296
  - Argumente weiterleiten 280
  - Arrays von Objekten 299
  - Aufruf 297
  - Aufruf in Funktionsaufrufen 297
  - Aufruf mit new 268, 297
  - Basisklassenkonstruktoren 314
  - Definition 296
  - Ersatzkonstruktor 298
  - Konstruktortabelle 279
  - Kopierkonstruktur 378, 380
  - Standardkonstruktoren 298
  - Vererbung 314
- Kontrollstrukturen 93
  - Bedingungen 71f.
  - do-while-Schleife 88
  - for-Schleife 89, 129
  - goto 96
  - if-Anweisung 76
  - if-else-Verzweigung 78
  - Schleifen 84
  - Sprunganweisungen 92
  - switch-Verzweigung 81
  - Verzweigungen 75
  - while-Schleife 84
- Konvertierungen
  - automatische 355
  - Integral Promotion 355
  - Zahlen in Strings 201
  - Zeit in String 225

- Kopieren
  - flache Kopien 378
  - Objekte 378
  - tiefe Kopien 380
 Kopierkonstruktur 378, 380
- L**
- Lambda-Ausdrücke (C++11) 401
  - auto-Variable 401
  - Parameter 402
  - Rückgabewert 402f.
  - Syntax 401
  - Zugriff auf umgebende Variable 402
 Laufzeitfehler 173
 Laufzeitmessung 229
 Laufzeitmessung (C++11) 230
 Laufzeittypidentifizierung 333
  - dynamic\_cast 333
  - typeid 335
 Lebensdauer 385
  - Objekte 385
  - Variablen 385
 leere Anweisungen 97
 length() (string) 199
 Linker 18
 list (Container-Klasse) 241, 249
 Listen 236
 Literale 39
  - Ganzzahlen 40
  - Gleitkommazahlen 41
  - Strings 39, 191
  - Strings (C++11) 194
  - Verwendung 41
 localtime() 225
 Lochkartensysteme 4
 logic\_error (Klasse) 345
 long 53, 436
 long long (C++11) 53, 353, 436
 L-Wert 48
- M**
- main() 23, 114
  - argc 114
  - argv 114
 Maschinenbefehle 5
 Mathematische Funktionen 64, 183
 Mehrfachvererbung 319
 Memberfunktionen 142, 288
  - abstrakte 336
  - Basisklassenparameter 332
  - Bindung 327
  - Definition 288
- Deklaration 272
 - Function overhead 289
 - generische 332
 - get/set 293
 - inline 276, 289
 - konstante 292
 - override (C++11) 326
 - rein virtuelle 336
 - statische 291
 - this-Zeiger 290
 - Überladung 318
 - Überschreibung 318
 - virtuelle 328
- Membervariablen 138, 142, 277
  - Definition 277
  - Initialisierung 278
  - Initialisierung (C++11) 280
  - Instanzvariablen 278
  - Klassenvariablen 255, 287
  - konstante 286
  - mutable 293
  - private-Deklaration 282
  - von Klassentypen 284
 Mengen 236
 Menüs, für Konsolenanwendungen 82
 Modularisierung 6, 103
  - Funktionen 254
  - Klassen 254
  - Typdefinitionen 256
 Modulo 63
 MS-DOS-Eingabeaufforderung 25, 36
 mutable 293
- N**
- Namen
  - Groß- und Kleinschreibung 44
  - Regeln für die Namensgebung 44
  - vollqualifizierte 16
 Namensgebung 44
 Namenskonflikte 15
  - this-Zeiger 291
  - Verdeckung 317
  - Namensräume 15
  - Namenskonflikte 15
  - Standardbibliothek 15f.
  - std 15f.
  - vollqualifizierte Namen 16
 namespace 16
 Namespaces *siehe* Namensräume
 Nebeneffekte 98
 Neue Zeile-Zeichen 192
- new 160
 new[] 164
 noexcept (C++11) 346
 NULL 149
 nullptr 149
 numeric\_limits 354
- O**
- Objekte
  - Arten 263
  - Basisklassenobjekt 299
  - eingebettete 284
  - Elementzugriff 268
  - Instanzen 267
  - Instanzierung 266, 297
  - Instanzvariablen 278
  - Klassen 265
  - Konstruktor 267
  - kopieren 378
  - Lebensdauer 385
  - Punktoperator 268
  - vergleichen 374
  - Zeiger 268, 297
  - Zugriff auf Elemente 270
  - Zustand 278
 Objektorientierte Programmierung 261
  - Ausnahmebehandlung 339
  - Instanzierung (Objekterzeugung) 266
  - Klassen 261, 265
  - Konstruktor 267
  - Objekte 266
  - Polymorphie 323
  - Softwareentwicklungszyklus 264
  - Vererbung 305
 Objektorientiertes Denken 262
 Operandenkonvertierung 355
 Operatoren 61
  - \
  - \ (Gültigkeitsbereichsoperator) 16, 119
  - + (Addition) 62
  - + (Addition, string) 68, 195f.
  - & (Adresse) 148
  - arithmetische 61
  - Bit-Manipulation 356
  - ^ (bitweises Komplement) 356
  - ~ (bitweises Komplement) 357
  - | (bitweises ODER) 356
  - & (bitweises UND) 356
  - const\_cast 59

- << (cout) 203
  - -- (Dekrement) 67
  - Dekrement 67
  - delete 160, 301
  - delete[] 164
  - Dereferenzierung 150
  - / (Division) 62
  - Division 63
  - dynamic\_cast 59, 333
  - >> (Eingabe, cin) 210
  - == (Gleichheit) 73
  - === (Gleichheit, string) 197
  - > (Größer) 73
  - >= (Größer oder gleich) 73
  - >= (Größer oder gleich, string) 197
  - > (Größer, string) 197
  - Gültigkeitsbereichsoperator 16, 119
  - [] (Indizierung) 126
  - [] (Indizierung, vector) 239
  - ++ (Inkrement) 67
  - Inkrement 67
  - ++ (Inkrement, Iteratoren) 242
  - < (Kleiner) 73
  - <= (Kleiner oder gleich) 73
  - <= (Kleiner oder gleich, string) 197
  - < (Kleiner, string) 197
  - kombinierte Zuweisungen 66
  - << (Linksverschiebung) 357
  - logische 74
  - ! (logisches Nicht) 75
  - || (logisches ODER) 75
  - && (logisches UND) 74
  - . (Memberzugriff) 268
  - % (Modulo) 62
  - Modulo 63
  - \* (Multiplikation) 62
  - Nebeneffekte 98
  - new 160
  - new[] 164
  - Priorität 66
  - >> (Rechtsverschiebung) 357
  - reinterpret\_cast 59
  - relationale (Vergleichsoperatoren) 73
  - sizeof 128
  - static\_cast 59
  - - (Subtraktion) 62
  - typeid 335
  - Typumwandlung 58
  - != (Überladung) 374
  - + (Überladung) 372
  - ++ (Überladung) 371
  - += (Überladung) 372
  - < (Überladung) 376
  - << (Überladung) 373
  - <= (Überladung) 376
  - = (Überladung) 380
  - == (Überladung) 374
  - > (Überladung) 376
  - >= (Überladung) 376
  - >> (Überladung) 373
  - Überladung 370
  - != (Ungleichheit) 73
  - != (Ungleichheit, string) 197
  - + (Vorzeichen) 62
  - + (Vorzeichen) 62
  - Optimierung 175
  - Compiler 176
  - Laufzeitmessung 229
  - Schleifen 91
  - ostream (Klasse) 203
  - << 203
  - fill() 205
  - precision() 205
  - width() 204
  - override (C++11) 326
  - printf() 206
  - private 270, 282, 317
  - Programme
  - Ablaufsteuerung 71
  - Algorithmus 11
  - auf Drücken der Eingabetaste warten 37
  - Beispielprogramme auf Buch-DVD 427
  - Daten einlesen 210
  - debuggen 174
  - Ergebnisse ausgeben 203
  - Erstellung 28
  - Laufzeitmessung 229
  - main() 114
  - optimieren 175
  - Programmausführung 35
  - Programmgerüst 21
  - Programmausführung 35
  - Programmerstellung 10, 17, 28
  - Ablauf 10, 17
  - mit dem GNU-Compiler 34
  - mit Visual Studio 28
  - objektorientierte Problemlösung 264
  - Programmierung
  - Assembler 6
  - Geschichte 4
  - Lochkartensysteme 4
  - Modularisierung 6
  - objektorientierte 8
  - strukturierte 6
  - protected 270, 312, 317
  - public 270, 317
  - push\_back() (vector) 239
  - push\_front() (deque) 419
- P**
- pair (Klasse) 250
  - Parameter 111
  - call by reference 154, 157
  - call by value 112
  - const 158
  - definieren 111
  - Referenzen 154
  - von Basisklassentypen 332
  - Vorgabeargumente 113
  - Zeiger 154
- Passwortabfragen 77
- Passwörter 78
- peek() (cin) 238
- Polymorphie 323
- abstrakte Basisklassen 335
  - Basisklassenarrays 330
  - Basisklassenparameter 332
  - Bindung 327
  - Definition 323
  - Grundprinzip 323
  - Typidentifizierung 333
  - Überschreibung 325
- pop\_back() (vector) 240
- pop\_front() (deque) 419
- Präprozessor
- Compiler-Schalter 256
  - Headerdateien 15
- precision() (ostream) 205
- Primfaktorzerlegung 359
- Primzahlen 360
- R**
- Radiant 186
  - RAM 42
  - rand() 187
  - rdstate() (basic\_ios) 212
  - Rechenoperationen 61
  - Referenzen 152
  - const 157
  - Definition 153
  - Initialisierung 153
  - Parameter 154
- Reguläre Ausdrücke (C++11) 393
- Assertionen 397
  - Ersetzen 399
  - Gruppierung 396
  - Musterabgleich 397
  - Quantifizierer 395

- Suchen 398
- Syntax 393
- Zeichenklassen 394
- `reinterpret_cast` 59
- Rekursion 363
- `replace(string)` 200
- `reserve(vector)` 239
- `resizel(vector)` 239
- `return` 96, 109
  - Funktionen verlassen 110
  - Rückgabewert 109
- `rfind(string)` 200
- RSA 361
- Rückgabewerte 109
- R-Wert 48
  
- S**
- Schleifen 84
  - Abbruchbefehl 93
  - Arrays durchlaufen 128
  - Ausführung 86
  - do-while 88
  - Endlosschleifen 94, 100
  - for 89, 129
  - gewollte Endlosschleife 95
  - Iteration 86
  - leere Schleifen 97
  - mit mehreren Schleifenvariablen 91
  - Optimierung 91
  - reduzierte for-Schleife (C++11) 98
  - Schleifenvariable 86
  - while 84
- Schlüssel/Wert-Paare 248
- Semikolon 26, 88, 97
- short 53, 353, 436
- Sichtbarkeit 119
- Signaltonzeichen 192
- Signatur 121, 361
- `size()` (Container) 240
- `size()` (vector) 238
- `sizeof` 128
- Softwareentwicklungszyklus
  - Analyse 264
  - Anforderungsspezifikation 264
  - Design 264
  - Implementierung 265
  - Test 265
- Sonderzeichen
  - \a (Signalton) 192
  - \n (Zeilenumbruch) 192
  - \t (Tabulator) 192
  - Umlaute 207
- Sortieren
  - Arrays 248
- Speichermodelle
  - Heap 159
  - Stack 119
  - statischer Speicher 159
- Speicherverwaltung
  - Arrays 126
  - Ganzzahlen 50
  - Gleitkommazahlen 51
  - Zeichen 52
- Sprunganweisungen 92
  - für Funktionen 96
  - für Schleifen 93
- `srand()` 188
- Stack 119
  - lokale Variablen 119
  - stack unwinding 348
- Standardbibliothek 14, 18, 449
  - ANSI C 449f.
  - Ausnahmen 344
  - cin 210
  - complex 189
  - cout 203
  - Dateien 214
  - Datentypen 59
  - Ein- und Ausgabe 203
  - Headerdateien 15
  - ifstream 217
  - Iteratoren 240
  - komplexe Zahlen 189
  - mathemat. Funktionen 183
  - numeric\_limits 354
  - ofstream 214
  - rand() 187
  - reguläre Ausdrücke (C++11) 393
  - std-Namensraum 15f.
  - STL 233
  - Streams 213
  - string 195
  - Strings 191
  - Zeitnahme (C++11) 224
  - Zufallszahlen 187
  - Zufallszahlen (C++11) 189
  - Zugriff auf Elemente 16
- Standardkonstruktoren 298
- Stapel 236
- Starten, von Visual Studio 426
- static
  - Memberfunktionen 291
  - Membervariablen 255, 287
- `static_cast` 59
- Statischer Speicher 159
- Stil 37
- STL (Standard Template Library) 233, 450
  - Algorithmen 243
  - Aufbau 233
  - Datenstrukturen 236
  - Iteratoren 240
- Streams 213
- `strftime()` 226
- string (Klasse) 195
  - != 197
  - [] 199
  - + 196
  - < 197
  - <= 197
  - == 197
  - > 197
  - >= 197
  - clear() 199
  - compare() 197
  - c\_str() 199
  - erase() 199
  - find() 199
  - find\_first\_of() 199
  - find\_last\_of() 199
  - insert() 199
  - length() 199
  - replace() 200
  - rfind() 200
  - substr() 200
- Strings 26, 39, 191
  - C-Strings 200
  - erzeugen 195
  - Escape-Sequenzen 192
  - \ (Escape-Zeichen) 192
  - Klasse string 195
  - + (Konkatenation) 68, 195
  - konkatenieren 196
  - Literale 39, 191
  - Literale (C++11) 194
  - reguläre Ausdrücke (C++11) 393
  - Tabulatoren 192
  - umbrechen 194
  - Umlaute 207
  - Umwandlung 201
  - vergleichen 74, 196
  - vergleichen (nach dt. Alphabet) 198
  - Zeilenumbruchzeichen 192
- stringstream (Klasse) 201
  - << 201
  - >> 201
  - clear() 201
- Stroustrup, Bjarne 8
- struct 138, 142
- Strukturen 137
  - Arrays von Strukturen 140
  - Definition 138, 142
  - initialisieren 140

- kontra Klassen 145
- Variablen 139
- Zugriff auf Elemente 139
- substr() (string) 200
- switch 81
- switch-Verzweigung 81
- break 81
- case-Marken 81
- default-Marke 81
- Fallthrough 83
- mit char-Ausdruck 82
- mit int-Ausdruck 82
- Syntaxreferenz 435

## T

- Tabulatorzeichen 192
- Tauschproblem 155
- template 388
- Templates 180, 387
  - Funktionen-Templates 388
  - Klassen-Templates 389
  - STL 233
- Text *siehe* Strings
- this (Instanzzeiger) 290
- throw 345ff.
- time() 223
- tm (Struktur) 224f.
- trigonometrische Funktionen 186
- true 53
- try 342
- typedef 241
- Typedefinitionen
  - Modularisierung 256
  - über Dateigrenzen hinweg verwenden 256
- typeid 335
- Typidentifizierung 333
  - dynamic\_cast 333
  - typeid 335
- Typumwandlung 55
  - cin 55
  - cout 55
  - dynamic\_cast 333
  - Integral Promotion 355
  - Operatoren 58
  - Zahlen 57

## U

- Überladung 121
  - != 374
  - + 372
  - ++ 371
  - += 372
  - < 376

- << 373
- <= 376
- = 380
- == 374
- > 376
- >= 376
- >> 373
- arithmetische Operatoren 372
- Funktionen 121
- Memberfunktionen 318
- Namensauflösung 122
- Operatoren 370
- Streamoperatoren 373
- Vergleichsoperatoren 374, 376
- Zuweisungsoperator 380
- Überlaufverhalten 354
- Überschreibung 318, 325f.
- Übersetzungseinheit 18, 256
- Umlaute 207
- unsigned 353
- using 17

## V

- Variablen 42
  - const 48
  - Darstellung im Arbeitsspeicher 49
  - Datentypen 42, 49, 53
  - definieren 42
  - Deklaration 255
  - globale 116
  - Gültigkeitsbereiche 118
  - initialisieren 46
  - Lebensdauer 385
  - lokale 116
  - Typumwandlung 55
  - über Dateigrenzen hinweg verwenden 255
  - Werte abfragen 47
  - Werte zuweisen 45
- vector (Container-Klasse) 237
- Verdeckung 119, 317, 384
- Vererbung
  - abgeleitete Klassen 305
  - abstrakte Basisklassen 335
  - Basisklassen 305
  - Basisklassenunterobjekte 299, 310
  - Destruktor 319
  - Grundprinzip 305
  - Konstruktoren 314
  - kontra Komposition 308
  - Mehrfachvererbung 319
  - private Elemente 421
  - Sinn 308

- Syntax 307
- Überladung 318
- Überschreibung 318, 326
- Verdeckung 317
- Zugriffsspezifizierer 312, 316
- Vergleiche
  - Objekte 374
  - Operatoren 73
  - Strings 74, 196
  - Strings (nach dt. Alphabet) 198
- Verschachtelung
  - 'dangling else' 99
  - if-Anweisung 80
- Verschlüsselungsverfahren 361
- Verzweigungen 75
  - 'dangling else' 99
  - if-Anweisung 80
  - if-else-Verzweigung 78
  - switch-Verzweigung 81
- virtual 325, 328, 336
  - Destruktor 319
  - rein virtuelle Memberfunktionen 336
  - Überschreibung 319
- Visual Studio 28
  - Befehlszeilenargumente 115
  - Installation 425
  - Programme ausführen 33
  - Programme erstellen 32
  - Projekte anlegen 29
  - Projektmappen 30
  - Quelldateien 31
  - starten 426
  - void 110
- Vorgabeargumente 113
- Vorkomplizierte Header 30
- Vorzeichen 62

## W

- Warteschlangen 236
- Warteschleifen 231, 418
- what() (exception) 345
- while 84
- Whitespace 219
- width() (ostream) 204
- Windows-Konsole (Eingabe-aufforderung) 25, 36
- write() 220

## Z

- Zahlen
  - Binärzahlen 40
  - Division 63
  - Ganzzahlen 40, 50

- Gleitkommazahlen 41, 51, 205
- Hexadezimalzahlen 40
- komplexe 189
- mathematischen Funktionen 64, 183
- Modulo 63
- Rechenoperationen 61
- trigonometrische Funktionen 186
- Typumwandlung 57
- Überlaufverhalten 354
- Umwandlung in Strings 201
- Vergleichen 73
- Vorzeichen 62
- Zufallszahlen 187
- Zähler 76
- Zeichen 52
- Zeichensätze
  - ANSI 432
  - ASCII 431
  - Unicode 192
- Zeiger 147
  - Arithmetik 152
  - auf Funktionen 361
  - auf Memberfunktionen 367
  - auf Objekte 268, 297
  - const 157
  - Definition 148
  - Dereferenzierung 150
  - dynamische Speicherreservierung 159
  - Initialisierung 148
  - nullptr (NULL) 149
  - Parameter 154
  - this (Instanzzeiger) 290
- Zeilenumbrüche 194
- Zeit und Datum 229
  - abfragen 223
  - aufschlüsseln 224
  - C++11 224
  - Greenwich Mean Time 224
  - in String formatieren 226
- in String konvertieren 225
- tm-Struktur 225
- Warteschleifen 231, 418
- Zeitspanne 230
- Zeitmessung 229
- Zeitspanne 230
- Zufallszahlen 187
- Zufallszahlen (C++11) 189
- Zugriffsspezifizierer 269
  - bei Vererbung 316
  - friend 369
  - private 270, 282, 317
  - protected 270, 312, 317
  - public 270, 317
- Vererbung 312
- Zuse, Konrad 6
- Zustand, eines Objekts 278
- Zuweisungsoperator 378, 380