

# Register

## Symbole

\* 46, 202, 441  
\*= 46, 360  
+, +=, -, -= 46  
++ 46, 362, 437, 441  
, 61, 81, 486  
->\* 249  
-- 46, 365  
.\* 249  
... siehe Ellipse  
, /= 46  
/\* ... \*/ 34  
// 34  
:: 62, 164, 293  
::\* 249  
; 36, 71  
<, <=, >, >= 46, 57  
<< 46, 103, 243, 349, 407  
<<=, >>= 46  
=, == 46, 57, 58  
>> 46, 101, 243, 411, 428  
?: 71  
[ ] 206, 209, 354  
[ ][ ]  
    Matrixklasse 391  
    Zeigerdarstellung 232  
# 141  
%, %= 46

& Adress-Operator 59  
&, &= Bit-Operatoren 46  
&& logisches UND 58  
&& Shell 608  
&& R-Wert 454  
\ 34, 55, 139, 338  
\0 210, 214–216  
\", \a, \b 55  
\f, \n, \r, \t, \v 55, 102  
\x 55  
\\ 55  
~ 46  
^ 46  
|= 46  
|| 58  
! 46, 58  
!= 57, 58, 441  
\$<, \$^, @\$ 605  
@D, @F 613  
" 36, 210, 211

## A

abgeleitete Klasse 279, 288, 292  
    und virtueller Destruktor 305  
Abhängigkeit (make) 603  
    automatische Ermittlung 608  
abort() 937  
abs() 52, 742, 743, 923, 935, 936

- Abstrakter Datentyp 160, 963
- abstrakte Klasse 299, 963
- accumulate() 695
- acos() 742, 923, 935
- acosh() 742
- Adapter, Iterator- 864
- Additionsoperator 347
- adjacent\_difference() 698
- adjacent\_find() 726
- adjustfield 409
- Adresse 202
  - symbolische 36
- Adressierung, indirekte (valarray) 931
- Adressoperator 203
- advance() 862
- Aggregat 838
- Aggregation 669, 963
- Aktualparameter 112
- <algorithm> 671, 761, 869
- Algorithmus 30, 437
  - accumulate() 695
  - adjacent\_difference() 698
  - adjacent\_find() 726
  - all\_of 707
  - any\_of 707
  - binary\_search() 728
  - clamp() 768
  - copy(), copy\_backward() 758
  - copy\_if() 760
  - copy\_n() 760
  - count(), count\_if() 706
  - equal() 741
  - equal\_range() 729
  - fill(), fill\_n() 693
  - find(), find\_if(), find\_if\_not() 721
  - find\_end() 725
  - find\_first\_of() 722
  - for\_each(), for\_each\_n() 758
  - generate(), \_n() 694
  - includes() 731
  - inner\_product() 696
  - inplace\_merge() 720
  - iota() 694
  - is\_heap(), is\_heap\_until() 738
  - is\_partitioned() 713
  - is\_permutation() 709
  - is\_sorted(), is\_sorted\_until() 714
  - iter\_swap() 761
  - lexicographical\_compare() 711
  - lower\_bound() 729
  - make\_heap() 737
  - make\_pair() 793
  - make\_tuple() 794
  - max(...) 767
  - max\_element() 699
  - merge() 718
  - mergesort() 719
  - min(...) 767
  - min\_element() 699
  - minmax() 767
  - minmax\_element() 700
  - mismatch() 739
  - move(), move\_backward() 791
  - next\_permutation() 708
  - none\_of 707
  - nth\_element() 717
  - partial\_sort(), \_copy 716
  - partial\_sum() 697
  - partition() 712
  - partition\_copy(), partition\_point() 712
  - pop\_heap() 736
  - prev\_permutation() 708
  - push\_heap() 737
  - remove(), \_if(), \_copy(), \_copy\_if() 764
  - replace(), \_if(), \_copy(), \_copy\_if() 763
  - reverse(), reverse\_copy() 704
  - rotate(), rotate\_copy() 701
  - sample() 705
  - search() 723, 725
  - search\_n() 727
  - set\_difference() 733
  - set\_intersection() 733
  - set\_symmetric\_difference() 734
  - set\_union() 732
  - shuffle() 702
  - sort() 714
  - sort\_heap() 738
  - split() 672
  - stable\_partition() 712

- stable\_sort() 715
- swap(), swap\_ranges() 761
- transform() 761
- unique(), unique\_copy() 702
- upper\_bound() 729
- Alias-Name 59, 203, 205
  - \*this 227
- alignment\_of 815
- alignof() 382, 815
- all\_of 707
- allgemeiner Konstruktor 168
- allocator 786
- alternative Funktions-Syntax 129
- Anführungszeichen 36, 210, 211
- Anker (bei regulärem Ausdruck) 492
- anonymer Namespace 136
- ANSI-Sequenzen zur Bildschirm-  
ansteuerung 55
- Anweisung 65
- any() (Bitset) 799
- any\_of 707
- app 423
- append()
  - Pfad 774
  - String 900
- apply()
  - Tupel 796
  - valarray 920
- Äquivalenzbeziehung 728
- Äquivalenzklasse 628
- arg() 742, 743
- argc 225
- Argument *siehe* Parameter
- Argument Dependent Lookup (ADL) 964
- argv[] 225
- Arität (Template) 480
- Arithmetik
  - mit Iteratoren 440
  - mit Zeigern 208
- arithmetische Operatoren 46
- Array
  - char 212
  - von C-Strings 213
  - dynamisches 218, 233, 234, 351
  - Freigabe 221
  - als Funktionsparameter 224, 230
  - mehrdimensionales 229, 233, 234
  - Matrixklasse 391
  - valarray 917
  - vs. Zeiger 207
- <array> 820, 838
- Array2d 235, 590
- ASCII 879
  - Dateien 242
  - Tabelle 56, 941
- asctime() 938
- asin() 742, 923, 935
- asinh() 742
- assert() 142
  - mit Exception 338
- assign() 825, 901
- Assoziation (UML) 666
- Assoziativität von Operatoren 60, 946
- async() 532
- at() 89, 93, 827, 832, 839, 845, 899
- atan() 742, 923, 935
- atan2() 924, 935
- atanh() 742
- ate 423
- atexit() 937
- atof(), atol() 937
- atoi() 937
- Atom-Uhr 569
- atomic 529
- Attribut
  - Compilersteuerung 947
  - einer Klasse 161, 964
- Aufforderung 31, 964
- Aufzählungstyp 84
- Ausdruck
  - Auswertung 60
  - Definition 42
  - mathematischer 52, 124
- Ausgabe 101, 103, 407
  - benutzerdefinierter Typen 407
  - Datei- 104, 405
  - Formatierung 407
  - Weite der 408
- Ausgabeoperator 349, 407
- Ausnahme 331, 964
  - behandlung 331
- Ausrichtung an Speichergrenzen 381

- Auswertungsreihenfolge 52, 60, 946
- auto 97, 150, 446, 650
- auto[] 98, 794
- automatische Variable 134
  - make 605
- Autotools (GNU) 626
- B**
- back() 826, 829, 832, 835, 838, 899
- back\_inserter(), \_insert\_iterator 865
- Backslash
  - Zeichenkonstante \ 56
  - Zeilenfortsetzung 139, 338
- Backspace 56
- bad() 421
- bad\_alloc 337
- badbit 420
- bad\_cast 320, 337
- bad\_function\_call 337
- bad\_typeid 323, 337
- bad\_weak\_ptr 337
- base() 864
- basefield 409
- basic\_string 897
- basic\_~Streamklassen 406
- Basisklasse 279
  - und virtueller Destruktor 305
  - Konstruktor 313
  - Subobjekt 317
  - virtuelle 315
- Bedingungsausdruck 68
  - make 619
- Bedingungsoperator ?: 71
- beg 424
- begin()
  - Container 442, 822
  - Namespace std 209, 863
  - string 257, 898
  - vector 91
- Belegungsgrad 850
- benutzerdefinierte
  - Datentypen 84, 87
  - Klassen 349
  - Literale 388
  - Typen (Ausgabe) 407
  - Typen (Eingabe) 428
- Benutzungszählung 912
- Bereichsnotation 823, 966
- Bereichsoperator :: 62, 164, 293
  - namespace 145
- Bibliothek
  - C 156, 933
  - C++ 784
- Bibliotheksmodul 132
  - dynamisch 619
  - statisch 618
- Bidirectional-Iterator 858
- Big Three/Five (Regel) 240, 467
- Bildschirmsteuerung mit ANSI-
  - Sequenzen 55
- Binärdatei 244
- binäre Ein-/Ausgabe 241
- binärer Operator 347
  - optimiert 462
- binäres Prädikat 703, 740, 870
- binäre Zahlendarstellung 47
- Binärzahl 44
- binary 107, 423
- binary\_search() 728
- bind 802
- Binden 132
  - dynamisches *siehe* dynamisches B.
  - statisches *siehe* statisches Binden
- Bit
  - feld 100
  - Operatoren 46, 47
  - Verschiebung 47
  - pro Zahl 42
- bitset 797
- bitweises ODER, UND, XOR 46
- Block 34, 35, 62, 63, 66, 193
  - und dyn. Objekte 221
- bool 58
- boolalpha 58, 408, 415
- Boost
  - is\_any\_of() 673
  - lexical\_cast() 675
  - split() 673
- Boost.Asio 561
- break 73, 82
- bsearch() 937
- Bstatic (Makefile) 618

- Bucket 850
- bucket() 852
- Byte 54
  - std::byte 86, 936
  - Reihenfolge 569
- C**
- C++-Schlüsselwörter 943
- C-Arrays 205
- C-Funktionen einbinden 158
- C-Header 933
- C-String 210
- call wrapper 514
- Callback-Funktion 246, 804
- canonical() (Filesystem) 774
- capacity() 827, 900
- capturing group 490
- case 73
  - lokale Variablen 74
- <cassert> 142, 933
- cast *siehe* Typumwandlung
- catch 332
- cbegin()
  - Container 822
  - Namespace std 863
  - string 899
- <cctype> 762, 934
- cdecl 252
- ceil() 935
- end()
  - Container 822
  - Namespace std 863
  - string 899
- cerr 101, 407
- char 54, 208
- char\* 210
- char16\_t, char32\_t 880
- char\* const vs. const\* char 205
- char16\_t, char32\_t 387
- <chrono> 502
- cin 36, 101, 407, 421
- clamp() 768
- class 162, 287
- class (bei Template-Parametern) 151
- clear() 421, 825, 841, 902
- Client-Server
  - Beziehung 164
  - und callback 246
- clock(), clock\_t 938
- clog 101, 407
- close() 105
- CMake 625
- <cmath> 52, 785, 933, 934
- code bloat 622
- collate 680, 884
- combine() 878
- compare() 884, 904
- Compilationsmodell 623
- Compiler 33, 34, 37, 131, 161, 163
  - befehle 944
  - direktiven 130
  - und Templates 154
  - Typumwandlung 185
- <complex> 743
- Computerarithmetik 52
- conditional 817
- configure 626
- conj() 742
- connect() 539
- const 53
  - Elementfunktionen 163, 165
  - globale Konstante 135
- const\_pointer 841
- const\_cast<>() 321
- const char\* vs. char\* const 205
- constexpr
  - Funktion 147
  - Konstante 53
  - Konstruktor / Methode 180
- const\_iterator 821
- const\_local\_iterator 852
- const\_pointer 826
- const\_reference 821
- const\_reverse\_iterator 824, 865
- const& *siehe* Referenz auf const
- constraint, Vererben von 310
- Container 436
  - implizite Datentypen 821
- Container-Methoden 822
- container\_type 833, 834
- continue 82
- copy elision 451

- copy semantics 455
- copy() (Filesystem) 776
- copy() 758, 899
- copy\_backward() 758
- copy\_if() 760
- copy\_n() 760
- copysign() 935
- cos(), cosh() 742, 923, 935
- count() 799, 842
  - Algorithmus 706
- count\_if() 706
- cout 36, 101, 407
- \_\_cplusplus 158
- crbegin()
  - Container 824
  - Namespace std 863
  - string 899
- create\_directory() 777
- crend()
  - Container 824
  - Namespace std 863
  - string 899
- critical section 511
- CRLF 572
- cshift() 920
- <cstdlib> 48, 204, 936
- <cstdliblib> 124, 245
- c\_str() 899
- <cstring> 211, 247, 255, 258, 939
- <ctime> 364, 937
- ctype 679, 681, 884, 885
- cur 424
- curr\_symbol 889
- current\_path 774
- <cwctype> 934
- CXXFLAGS 605
- D**
- dangling pointer 220
- data race 511, 518
- data() 826, 838, 899
- Datagramm 567
- \_\_DATE\_\_ 143
- date\_order() 891
- Datei
  - ASCII 242
  - binär 244
  - Ein-/Ausgabe 104, 405
  - kopieren 106, 776
  - löschen 774
  - Öffnungsarten 423
  - öffnen 107
  - Positionierung 423
  - schließen 105
  - umbenennen 777
  - Zugriffsrechte 771
- Daten
  - als Attributwerte 965
  - static-Element- 265
- Datenbankanbindung 587
- Datenkapselung 965
- Datentypen 36, 42
  - abstrakte *siehe* Abstrakter Datentyp
  - benutzerdefinierte 84
  - int und unsigned 71
  - logische 58
  - parametrisierte 151, 271
  - polymorphe 303
  - strukturierte 87
  - zusammengesetzte 84
- Datum
  - Klasse / Gültigkeit 362
  - regulärer Ausdruck 750
- dec 408, 415
- decay, decay\_t 816
- decimal\_point() 887, 888
- decltype 275
- decltype(auto) 278, 693, 790
- declval 278
- default (in switch-Anweisung) 73
- default constructor 167
- Default-Parameter *siehe* vorgegebene P.
- default\_order 817
- default\_delete<X[]> 656
- = default 176
- #define 137, 139
- Definition 132, 965
  - von static-Elementdaten 265
  - von Objekten 168
- Deklaration 35, 36, 132, 965
  - einer Funktion 111

- Funktionszeiger 244
- Lesen einer D. 250
- in for-Schleifen 79
- Deklarationsanweisung 65
- Dekrementierung 46, 47
- Dekrementoperator 365
- Delegation 326
- delegierender Konstruktor 179
- delete 217, 220, 268, 304, 306
  - überladen 375
- delete [ ] 221
- = delete 176, 652
- <deque> 820, 831
- deque 831
- Dereferenzierung 202, 246
- Destruktor 192, 314, 467
  - implizite Deklaration 192
  - und exit() 194
  - virtueller 304
- detach() 508
- Dezimalpunkt 49, 887
- Dialog 550
- diamond problem 317
- difference\_type 821
- Differenz (Menge) 733
- difftime() 938
- digits, digits10 766
- distance() 862
- Distribution 607, 965
- div(), div\_t 936
- divides 801
- Division durch 0 340
- dll 619
- DNS 559
- do while 77
- domain\_error 337
- double 49, 208
  - nicht als Laufvariable 81
  - korrekter Vergleich 648
- downcast 320, 398
- Drei (die großen Drei) 240, 467
- Dubletten entfernen 702
- Durchschnitt (Menge) 733
- DYLD\_LIBRARY\_PATH 621, 962
- dynamic\_cast<>() 320
- dynamic\_pointer\_cast 913

- dynamisches Array 351
- dynamisches Binden 244, 293, 965
- dynamische Datenobjekte 217
- dynamischer Typ 322

## E

- e, E 49
- Editor 32
- effizienter binärer Operator 462
- egrep 489
- Ein- und Ausgabe 101
- Einbinden von C-Funktionen 158
- Eingabe 410
  - aufforderungsfenster 33
  - Datei- 104, 405
  - von Strings 103
  - benutzerdefinierter Typen 428
- Einschränkung *siehe* constraint
- Elementdaten, Zeiger auf 250
- Elementfunktion 160, 282
  - als Funktionsobjekt 805
  - Zeiger auf 249
- Ellipse
  - in catch-Klausel 334
  - template parameter pack 481
- else 67
- #else, #elif 137
- emplace() 823, 833, 835, 837
- emplace\_back() 827, 829, 832
- emplace\_front() 829, 832
- empty() 822, 833, 835, 836, 899
- enable\_if 816
- enable\_if\_t 817
- end 424
- end()
  - Container 442, 822
  - Namespace std 209, 863
  - string 257, 898
  - vector 91
- #endif 139
- endl 43, 55, 415
  - oder '\n'? 104
- ends 415
- ENTER** 34, 102
- »enthält«-Beziehung 315
- enum 85

- Environment, env[] 225
  - EOF 412
  - eof() 332, 412, 421, 438
  - eofbit 420
  - epsilon() 766
  - equal() 741
  - equal\_to 802
  - equal\_range() 729, 842
  - equalsIgnoreCase() 681
  - erase() 825, 841, 902
  - ereignisgesteuerte Programmierung 536
  - errno 330
  - error\_code 772
  - Exception 964
    - arithmetische Fehler 340
    - und Destruktor 331
    - Handling 331
    - Hierarchie 335
    - Speicherleck durch Exc. 654
  - <exception> 337
  - exception 335
  - Exception-Sicherheit 342
  - ExecutionPolicy 770
  - exists() (Filesystem) 775
  - exit() 124, 937
    - und Destruktor 194
  - Exklusiv-Oder (Menge) 734
  - exp() 742, 923, 935
  - explicit 176
  - explizite Instanziierung von
    - Templates 624
  - Exponent 49–51
  - Expression Templates 748
  - extension() (Filesystem) 774
  - extern 134, 135
  - extern "C" 158
  - extern template 623
  - external linkage 136
- F**
- f, F 49
  - Fünf (die großen Fünf) 467
  - fabs() 742, 935
  - Facette 884
  - fail() 421, 422
  - failbit 420, 429
  - fakultaet() 110
  - [[fallthrough]] 74
  - Fallunterscheidung 72
  - false 58
  - false\_type 811
  - falsename() 887
  - Fehlerbehandlung 329
    - Ein- und Ausgabe 420
  - Fibonacci 480, 699
  - \_\_FILE\_\_ 143
  - filename() 774
  - Filesystem
    - canonical() 774
    - copy() 776
    - create\_directory() 777
    - current\_path() 774
    - exists() 775
    - extension() 774
    - filename() 774
    - is\_directory() 775
    - recursive\_directory\_iterator 779
    - remove() 774
    - remove\_all() 775
    - stem() 774
  - fill() 408, 693
  - fill\_n() 693
  - final 652
  - find() 841
    - Algorithmus 721
    - string 903
  - find\_end() 725
  - find\_first\_of() 722
  - find\_...-Methoden (string) 904
  - findstring 619
  - fixed 408, 410, 415
  - Fixture 636
  - flache Kopie 238
  - flags() 409
  - flip() 798, 828
  - float 49
  - floatfield 409
  - floor() 935
  - flush() 410
  - flush (Manipulator) 415
  - fmod() 935
  - fmtflags 408



- Fold-Expression 484
  - for 78
    - Kurzform 96
  - foreach (make) 614
  - foreach(), for\_each\_n() 758
  - Formalparameter 112
  - Formatierung der Ausgabe 407
  - forward() 791
  - Forward-Iterator 858
  - <forward\_list> 820
  - forward\_list 786
  - frac\_digits() 889
  - Fragmentierung (Speicher) 222
  - Framework 537
  - free store 217
  - free() 378
  - frexp() 935
  - friend 261
  - front() 826, 829, 832, 835, 838, 899
  - front\_inserter(), \_insert\_iterator 865
  - <fstream> 105
  - fstream 406, 424
  - Füllzeichen 408
  - function 804
  - <functional> 337, 801
  - Funktion 110
    - mit Gedächtnis (static) 113
    - mit initializer\_list 823
    - klassenspezifische 265
    - mathematische 52, 934
    - vorgegebene Parameterwerte 121
    - Parameterübergabe
      - per Referenz 119
      - per Wert 115
      - per Zeiger 222
    - rein virtuelle 299
      - mit Definition 300
    - static 265
    - alternative Syntax 129
    - Überschreiben
      - in abgeleiteten Klassen 292
      - virtueller Funktionen 297
    - virtuelle *siehe* virtuelle Funktionen
  - Funktionsobjekte 373, 416
    - function 804
    - mem\_fn 805
  - Funktions-Template 151
  - Funktork *siehe* Funktionsobjekte
  - <future> 337
- ## G
- g++ 38
  - Ganzzahlen 42
  - garbage collection 222
  - gcd() 769
  - gegenseitige Abhängigkeit von Klassen 199
  - Genauigkeit 50
  - Generalisierung 280
  - generate(), generate\_n() 694
  - generische Programmierung 436
  - GET (http) 573
  - get() 102, 243, 411, 412
  - getaddrinfo() 560
  - getenv() 937
  - getline() für Strings 103, 905
  - getline(char\*,...) 412
  - getloc() 434
  - get\_money() (Manipulator) 414
  - get\_monthname() 891
  - getnameinfo() 560
  - get\_time() (Manipulator) 414
  - get\_time(), get\_weekday(), get\_year() (locale) 891
  - ggT 76
    - std::gcd() 769
    - schnell 187
  - Gleichheitsoperator bei Vererbung 399
  - Gleichverteilung 754
  - Gleitkommazahl 53
    - Syntax 49
  - global 62, 63
    - Namensraum 157
    - Variable 134, 135
  - gmtime() 938
  - GNU Autotools 626
  - good() 421
  - goodbit 420
  - Grafische Benutzungsschnittstelle 535
  - greater, greater\_equal 802
  - greedy (regex-Auswertung) 492

Grenzwerte von Zahltypen 766  
 Groß- und Kleinschreibung 34, 39  
 größter gemeinsamer Teiler *siehe* ggT  
 grouping() 887, 889  
 gsllice 927  
 gsllice\_array 930  
 guard (Threads) 512  
 Gültigkeitsbereich 123  
   Block 62  
   Datei 136  
   Funktion 112  
   Klassen 162  
   und new 220

## H

hängender Zeiger *siehe* Zeiger,  
   hängender  
 hardware\_concurrency() 503  
 has\_sort\_function 812  
 has\_facet() 878  
 hash() 885  
 Hash-Funktion 850, 851  
 hasher 852  
 has\_infinity 766  
 \_\_has\_include 138  
 has\_value() (optional) 809  
 »hat«-Beziehung 669  
 Header 36  
   C++-Standard 156  
   der Standardbibliothek 784  
 Header (Http) 572  
 Header-Datei 130  
   Inhalt 133  
 Heap 735  
 hex 408, 415  
 Hexadezimalzahl 44  
 Host Byte Order 569

## I

-I Compileroption 137  
 iconv 883  
 IDE 37  
 Identität von Objekten 162, 966  
 IEC 60559, IEEE 754 50  
 if 67  
 if constexpr 484, 812

#if, #ifdef, #ifndef 137  
 ifeq 619  
 ifstream 105, 406  
 ignore() 412  
 imag() 742, 743  
 imbue() 876  
 Implementation 131  
   -Vererbung 324  
   -sdatei, Inhalt 133  
 implizite Deklaration  
   Destruktor 192  
   Konstruktor 167  
   Zuweisungsoperator 356, 396  
 in 423  
 #include 35, 130, 136  
 includes() 731  
 Indexoperator 89, 206, 209, 232, 354  
 index\_sequence 799  
 indirekte Adressierung (mit  
   indirect\_array) 931  
 infinity() 766  
 Initialisierung  
   array 838  
   direkte I. der Attribute 168  
   und virtuelle Basisklassen 317  
   C-Array 208, 229  
   einfacher Datentypen 45  
   mit Element-Initialisierungsliste 169  
   Konstante in Objekten 170, 266  
   globaler Konstanten 134, 135  
   Reihenfolge 271  
   mit {}-Liste 177, 823, 863  
   mit konstruktor-interner Liste 266  
   von Objekten 166  
   mit {}-Liste 168  
   von Referenzen 951  
   Reihenfolge 170  
   in for-Schleife 79  
   von static-Elementdaten 265  
   struct 87  
   und Vererbung 285  
   und Zuweisung 45, 172  
 initializer\_list 178, 823  
 Inklusions-Modell 623  
 Inkrementierung 46, 47  
 Inkrementoperator 362

- inline
  - Elementfunktion 165
  - Funktion 146
  - Konstante 135, 269
  - Variable 147, 269
- inner\_product() 696
- inplace\_merge() 720
- Input-Iterator 858
- insert() 825, 841, 849, 901
- insert\_iterator 866
- inserter() 866
- insert\_or\_assign 845
- Installation der Software 957
- Instanz 32, 966
- Instanziierung von Templates 273
  - explizite 624
  - ökonomische (bei vielen Dateien) 622
- int 35, 42
- int-Parameter in Templates 274
- intX\_t, int\_fastX\_t, int\_leastX\_t (X = 8, 16, 32, 64), intmax\_t 48
- integer\_sequence 799
- integral\_constant 811
- integral promotion 62
- Interface (UML) 664
- internal 408, 415
- internal linkage 136
- Internet-Anbindung 557
- Intervall (und Notation) 966
- invalid\_argument 337
- <iomanip> 414, 416
- <ios> 414
- ios 406, 420, 422
- ios\_base 406
- ios\_base::binary 107
- ios\_base-Fehlerstatusbits 420
- ios\_base-Flags 408, 409
- ios\_base-Manipulatoren 415
- ios::failure 421
- ios-Flags zur Dateipositionierung 424
- ios-Methoden 409, 410, 421
- iostate 420
- <iostream> 36, 406, 407, 411, 414
- iota() 694
- IPv4, IPv6 559
- IPv4-Adresse (regulärer Ausdruck) 752
- is() 886
- isalnum(), isalpha() 884, 934
- is\_any\_of() 673
- is\_arithmetic 813
- is\_base\_of 815
- isblank() 934
- is\_bounded 766
- is\_class 810
- iscntrl() 884, 934
- isdigit() 128, 174, 884, 934
- is\_directory() (Filesystem) 775
- is\_exact 766
- isgraph() 884, 934
- is\_heap(), is\_heap\_until() 738
- is\_iec559, is\_integer 766
- islower() 884, 934
- is\_modulo 766
- ISO 10646 880
- ISO 8859-1, ISO 8859-15 879
- is\_partitioned() 713
- is\_permutation() 709
- isprint() 884, 934
- ispunct() 884
- is\_same 815
- is\_signed 766
- is\_sorted(), is\_sorted\_until() 714
- isspace() 884, 934
- ist-ein-Beziehung 280, 289, 309
- istream 406, 410
- Istream-Iterator 866
- istream::seekg(), tellg() 423
- istream::ws 415
- istringstream 406, 425
- isupper() 884, 934
- isxdigit() 884, 934
- iter\_swap() 761
- Iterator 257, 437, 441, 857
  - Adapter 864
  - Bidirectional 858
  - Forward 858
  - Input 858
  - Insert 865
  - Output 858
  - Random Access 858

- Reverse 864
  - Stream 866
  - Tag 859
  - Zustand 441
  - <iterator> 857
  - iterator 821
  - iterator\_category 858
- J**
- Jahr 938
  - join() 506, 508
  - Jota 694
- K**
- Kardinalität 667
  - Kategorie (locale) 884
  - key\_equal 852
  - key\_type 841, 848
  - key\_comp(), key\_compare 842
  - value\_comp(), value\_compare 842
  - Klammerregeln 60
  - Klasse 31, 161, 966
    - abgeleitete *siehe* abgeleitete Klasse
    - abstrakte 299
    - Basis- *siehe* Basisklasse
    - Deklaration 163
    - konkrete 299
    - Ober- *siehe* Oberklasse
    - für einen Ort 162
    - Unter- *siehe* Unterklasse
    - für rationale Zahlen 183
  - Klassenname 323
  - klassenspezifische
    - Daten und Funktionen 265
    - Konstante 269
  - Klassen-Template 271
  - Klassifikation 280, 967
  - Kleinschreibung 34
  - kleinstes gemeinsames Vielfaches
    - siehe* lcm()
  - Kollisionsbehandlung 850
  - Kommandointerpreter 603, 606
  - Kommandozeilenparameter 225
  - Kommaoperator 61, 81, 486
  - Kommentar 34
  - komplexe Zahlen 742
  - Komplexität 972
  - Komposition 670
  - konkrete Klasse 299
  - Konsole 33
  - Konsole auf UTF-8 einstellen 879
  - Konstante 53
    - globale 134, 135
    - klassenspezifische 269
  - konstante Objekte 164
  - Konstruktor 163, 166
    - allgemeiner 168
    - implizite Deklaration 167
    - delegierender 179
    - erben 290
    - Kopier- *siehe* Kopierkonstruktor
    - vorgegebene Parameterwerte 169
    - Typumwandlungs- *siehe* Typumwandlungskonstruktor
  - Kontrollstrukturen 65
  - Konvertieren von Datentypen *siehe* Typumwandlung
  - Kopie, flache/tiefe 238
  - Kopieren
    - von Dateien 106
    - von Objekten 238
    - von Zeichenketten 215
  - Kopierkonstruktor 171, 354, 467
    - Auslassung durch Compiler 173
  - Kreuzreferenzliste 690
  - kritischer Bereich 511
  - Kurzform-Operatoren 46
- L**
- l, L 49
  - L-Wert 66, 206, 212, 354, 967
  - Länge eines Vektors 697
  - Lambda-Funktionen 469
  - LANG 876
  - late binding 244
  - Laufvariable 79, 80
  - Laufzeit 202
    - und Funktionszeiger 244
    - und new 217
    - und Polymorphie 293
    - Typinformation 322

- lcm() 769
  - LD\_LIBRARY\_PATH 621
  - ldd 620
  - ldexp() 935
  - ldiv(), ldiv\_t 936
  - left 408, 415
  - length() 899
  - length\_error 337
  - less, less\_equal 802
  - lexical\_cast() 675
  - lexicographical\_compare() 711
  - lexikografischer Vergleich 793, 822, 967
  - <limits> 43, 49, 766
  - \_\_LINE\_\_ 143
  - Linken 135, 156
    - dynamisches 619, 967
    - internes, externes 136
    - statisches 618, 967
  - Linker 37
  - linksassoziativ 60, 228
  - list 829
  - <list> 787, 820
  - Liste
    - Initialisierungs- 169, 266
    - Initialisierungs- (bei C-Arrays) 229
  - Liste (Klasse) 442
  - Literal 211, 967
    - benutzerdefiniert 388
    - String 387
    - Zahl- 53
    - Zeichen- 54, 880
  - load\_factor() 852
  - local\_iterator 852
  - <locale> 875
  - localtime() 364, 938
  - lock\_guard 512
  - log(), log10() 742, 923, 935
  - logic\_error 336, 337
  - logical\_and, !\_not, !\_or 802
  - logischer Datentyp 58
  - logische Fehler 338
  - logische Negation 58
  - lokal (Block) 62
  - lokale Objekte 205
  - long 42
  - long double 49
  - lower\_bound() 729, 843
  - lvalue *siehe* L-Wert
- ## M
- magic number 968
  - main() 34, 35, 123
  - MAKE 613
  - make 601
    - automatische Ermittlung von Abhängigkeiten 608
    - parallelisieren 617
    - rekursiv 612
    - Variable 605
  - Makefile 132, 603
  - make\_from\_tuple() 795
  - make\_heap() 737
  - make\_index\_sequence 799
  - make\_integer\_sequence 799
  - make\_pair() 793
  - make\_shared 655, 913
  - make\_tuple() 794
  - make\_unique 253, 466, 655, 911
  - Makro 139
  - malloc() 378
  - Manipulatoren 413
  - Mantisse 50
  - <map> 820, 843
  - mapped\_type 845
  - mask\_array 930
  - match\_results 496
  - mathematischer Ausdruck 52
  - mathematische Funktionen 934
  - Matrix
    - C-Array 229
    - C-Array, dynamisch 233
    - Klasse 235
    - Klasse, mit zusammenhängendem Speicher 744
    - Vektor von Vektoren 390
  - max() 766, 767, 920
  - max(initializer\_list<T>) 767
  - max\_size() 822
  - max\_bucket\_count() 852
  - max\_element() 699
  - max\_exponent, max\_exponent10 766

- max\_load\_factor() 853
  - [[maybe\_unused]] 947
  - mehrdimensionales Array 229
  - mehrdimensionale Matrizen 390
  - Mehrfachvererbung 282, 312, 314
  - MeinString (Klasse) 255
  - mem\_fn() 805
  - member function *siehe* Elementfunktion
  - memcpy() 353
  - <memory> 337, 787
  - memory leak 221, 654
  - Mengenoperationen auf sortierten Strukturen 731
  - merge() 718, 830, 842
  - mergesort() 719
  - messages 884, 893
  - Metaprogrammierung 478
  - Methode 31, 32, 160, 968
    - Regeln zur Konstruktion von Prototypen 645
  - MIME 968
  - min() 766, 767, 920
  - min(initializer\_list<T>) 767
  - min\_element() 699
  - min\_exponent, min\_exponent10 766
  - minmax(...) 767
  - minmax\_element() 700
  - minus 801
  - Minute 938
  - mischen 718
  - mismatch() 739
  - mktime() 938
  - modf() 935
  - modulare Gestaltung 129
  - Modulo 46
  - modulus 801
  - Monat 938
  - monetary 884, 888
  - money\_get 889
  - money\_punct 888
  - money\_put 890
  - Monitor-Konzept 522
  - move semantics 455
  - move() R-Wert 459
  - move(), move\_backward()
    - Container-Bereich 791
  - moving constructor 458
  - mt19937 702, 753
  - multimap 847
  - multiplies 801
  - Multiplikationsoperator 360
  - Multiplizität (UML) 667
  - multiset 849
  - mutable
    - Attribut 165
    - Lambda-Funktion 474
  - mutex 511
- ## N
- '\n'
    - oder endl? 104
    - und regex\_replace 498
  - Nachbedingung 129, 968
  - Nachkommastellen 49
    - precision 427
  - Name 39
    - einer Klasse 323
  - name() 323, 878
  - Namenskonflikte bei Mehrfachvererbung 314
  - Namenskonventionen 39
  - Namespace 64, 144
    - anonym 136
    - in Header-Dateien 142
    - Verzeichnisstruktur 611
  - namespace 35, 144
  - namespace std 64
  - narrow() 886
  - nationale Sprachumgebung 876
  - NDEBUG 142, 338
  - negate 801
  - Negation
    - bitweise 46, 48
    - logische 58
  - Negationsoperator (überladen) 422
  - negative\_sign() 889
  - neg\_format() 889
  - Network Byte Order 569
  - Netzwerkprogrammierung 557
  - neue Zeile 56, 67

- new 217, 220, 268
  - Fehlerbehandlung 341
  - überladen 375
- <new> 337, 915
- new\_handler 341
- new Placement-Form 915
- next() 862
- next\_permutation() 708
- noboolalpha 415
- [[nodiscard]] 947
- noexcept 334
- none() 799
- none\_of 707
- norm() 742, 743
- Normalverteilung 756
- noshowbase, -point, -pos 415
- noskipws 415
- not\_equal\_to 802
- nothrow 342
- notify\_one(), --all() 517, 662
- nounitbuf, nouppercase 415
- npos 898
- NRVO *siehe* RVO
- nth\_element() 717
- NTP – Network Time Protocol 569
- NULL 204, 936
- nullopt 809
- nullptr 204
- Null-Zeiger und new 342
- numeric 884, 887
- numeric\_limits 49, 766
- numerische Auslöschung 51
- numerische Umwandlung 673, 675, 906
- num\_get, num\_put 887
- NumeriertesObjekt (Klasse) 266
- numpunct 887
- O**
- O-Notation 972
- Oberklasse 279, 292, 968
  - erben von 282
  - Subobjekt einer 285
  - Subtyp einer 288
  - Zugriffsrechte vererben 286
- Oberklassenkonstruktor 282, 285
- object slicing 289
- Objekt 30, 161, 163, 969
  - code 37
  - dynamisches 217
  - als Funktions- 373
  - hierarchie 315
  - Identität *siehe* Identität von Objekten
  - Initialisierung 166
  - konstantes 164
  - orientierung 159
  - Übergabe per Wert 172
  - verkettete Objekte 219
  - verwitwetes 221
  - vollständiges 317, 972
- Objektcode 37
- oct 408, 415
- ODER
  - bitweises 46
  - logisches 58
- Öffnungsarten für Streams 423
- offsetof 936
- ofstream 105, 406
- Oktalzahl 44
- omanip 416
- one definition rule 133
- open() 105
- Open Source 969
- Operator
  - arithmetischer 46
  - binärer 347
  - Bit- 46
  - für char 57
  - als Funktion 346
  - Kurzform 47
  - für Literale 386
  - für logische Datentypen 58
  - Präzedenz 60, 945
  - relationale 46, 58, 789
  - Syntax 346
  - Typumwandlungs- 366
  - überladen (<<) 407
  - überladen (>>) 428
  - unärer 347
  - für ganze Zahlen 46
- operator!() 422
- operator delete() 375

- operator new() 375
- operator string() 366
- operator()() 373
- operator\*() 361, 367, 441
- operator\*=( ) 360
- operator++() 362, 364, 441
- operator++(int) 650
- operator+=( ) 349
- operator->() 367
- operator<<() 349, 407, 797, 798
- operator<<=( ) 798
- operator=( ) 358
- operator==( ) 441, 799
  - bei Vererbung 399
- operator>>() 411, 797, 798
- operator>>=( ) 798
- operator[]() 393, 396, 827, 832, 839, 859
- operator&() 797
- operator&=( ) 798
- operator^=( ) 798
- operator^() 797
- operator|() 797
- operator|=( ) 798
- operator~() 798
- operator!=( ) 441, 799
- Optimierung durch Vermeiden
  - temporärer Objekte 173
- optional 809
- Optionale Objekte 808
- Ort (Klasse) 162
- ostream 349, 406, 407
- Ostream-Iterator 866
- ostream::endl, ends, flush() 415
- ostream::seekp(), tellp() 423
- ostringstream 406, 425
- out 423
- out\_of\_range 337
- Output-Iterator 858
- overflow 45, 51
- overflow\_error 337
- override 302, 651
- P**
- pair 792
- Parameter
  - expansion 482
  - einer Funktion 111
  - Pack 481
  - übergabe
    - per Referenz 119
    - per Wert 115
    - per Zeiger 222
- parametrisierte Datentypen 151, 271
- »part-of«-Beziehung 669
- partial\_sum() 697
- partial\_sort(), partial\_sort\_copy 716
- partielle Spezialisierung von
  - Templates 858
- partition() 712
- partition\_copy(), partition\_point() 712
- path 771
- patsubst 607
- peek() 413
- perfect forwarding 791
- Performance 449
- Permutationen 708
- PHONY 604
- $\pi$  53, 181
- Placement new/delete 915
- plus 801
- POD (plain old data type) 969
- pointer 826, 841
- Pointer, smarte 367
- polar() 742
- polymorpher Typ 303, 322
- Polymorphismus 293, 969
- pop() 833, 835, 837
- pop\_back() 827, 829, 832
- pop\_front() 829, 832
- pop\_heap() 736
- portabel (Zeichensatz) 882
- pos\_format() 889
- Positionierung innerhalb einer Datei 424
- positive\_sign() 889
- POSIX 876
- POST (http) 577
- postcondition *siehe* Nachbedingung
- Postfix-Operator 362
- pos\_type 423
- pow() 742, 924
- Prädikat



- Algorithmus mit P. 870
  - binäres 703, 740, 870
  - unäres 706
- Präfix-Operator 362
- Präprozessor 130, 136, 608
- Präzedenz von Operatoren 60, 945
- precision() 410
- precondition *siehe* Vorbedingung
- prev() 862
- prev\_permutation() 708
- PRINT (Makro) 141
- printf() 480
- Priority-Queue 836
- private 163, 286
- private Vererbung 324
- Programm
  - ausführbares 37
  - Strukturierung 109
- Programmierrichtlinien 654
- Projekte 132
- protected 286
- protected-Vererbung 327
- Prototyp
  - Funktions- 110
  - einer Methode 162
  - Regeln zur Konstruktion 645
- ptrdiff\_t 936
- public 163, 282, 286
- push() 833, 835, 836
- push\_heap() 737
- push\_back() 827, 829, 832
  - vector 93
- push\_front() 829, 832
- put() 105, 243, 407
- putback() 412
- put\_money(), \_time() (Manipulator) 414

## Q

- qsort() 245, 937
- Qt 537
- QThread 554
- Quantifizierer 492
- Queue 834
- <queue> 787, 820, 834
- quoted() 414

## R

- R-Wert 66, 206, 967
  - Referenz 455
- race condition 511, 534
- radix 766
- RAII 512, 636, 655, 970
- <random> 702
- Random-Access-Iterator 858
- random\_device 753
- range\_error 337
- <ratio> 806
- Rationale Zahl
  - Klasse 183
  - Template std::ratio 806
- rbegin()
  - Container 824, 864
  - Namespace std 863
  - string 899
- rdstate() 421
- read() 241
- real() 742, 743
- Rechengenauigkeit 50, 81
- rechtsassoziativ 60, 228
- recursive\_directory\_iterator 779
- reelle Zahlen 49
- ref(), reference\_wrapper 514, 808
- reference
  - bitset 797
  - Container 821
  - vector<bool> 827
- reference collapsing rules 454
- reference counting 912
- Referenz 59, 127
  - auf Basisklasse 297
  - auf const 119, 171
  - auf istream 411, 428
  - auf Oberklasse 289, 296
  - auf ostream 350, 407
  - Parameterübergabe per 119
  - Rückgabe per 355
  - auf R-Wert 455
- Referenzsemantik 239, 449
- <regex> 337, 497
- regex\_iterator 496
- regex\_match() 497
- regex\_replace() 498, 685

- regex\_search() 498, 683
  - reguläre Ausdrücke 489
  - Reihenfolge
    - Auswertungs- 52, 60
    - der Initialisierung 170
    - umdrehen 704
  - rein virtuelle Funktion 299
    - mit Definition 300
  - reinterpret\_cast<>() 208, 241, 321
  - Rekursion 116
    - Template-Metaprogrammierung 478, 482
  - rekursiver Abstieg 124
  - rekursiver Make-Aufruf 612
  - rel\_ops 789
  - relationale Operatoren 46, 58, 789
  - remove()
    - Algorithmus 764
    - Datei/Verzeichnis 774
    - Liste 830
  - remove\_if() 764, 830
  - remove\_reference 790, 816
  - remove\_reference\_t 816
  - remove\_all() Dateien/Verzeichnisse 775
  - rename() Datei/Verzeichnis 777
  - rend()
    - Container 824, 864
    - Namespace std 863
    - string 899
  - replace() 763, 902
  - replace\_copy(), \_if(), \_copy\_if() 763
  - reserve() 827, 900
  - reset() 798
  - resetiosflags() 414
  - resize() 827, 830, 832, 900
  - return 36, 173
  - return value optimization *siehe* RVO
  - reverse() (list) 830
  - reverse(), reverse\_copy() 704
  - reverse\_iterator 824
  - Reverse-Iterator 864
  - Reversible Container 824
  - right 408, 415
  - »rohes« Stringliteral 387
  - rotate(), rotate\_copy() 701
  - round\_error(), round\_style() 766
  - RTTI 322
  - Rule of zero/three/five 467
  - runtime\_error 337
  - rvalue *siehe* R-Wert
  - RVO 173, 451, 463
- ## S
- sample() 705
  - scan\_is(), scan\_not() 886
  - Schleifen 75
    - und Container 95
    - do while 77
    - for 78
    - und Strings 213
    - Tabellensuche 91
    - terminierung 76
    - while 75
  - Schlüsselwörter 943
  - Schnittmenge 733
  - Schnittstelle 131, 970
    - einer Funktion 114
    - Regeln zur Konstruktion 645
  - scientific 408, 410, 415
  - scope 62
  - scoped locking 512
  - search() 723, 725
  - search\_n() 727
  - sed 615, 616
  - seekg(), seekp() 423
  - Seiteneffekt 68, 111, 214, 216
    - im Makro 143
  - Seitenvorschub 56
  - Sekunde 938
  - Selektion 67
  - sentinel 91, 209
  - sentry 434
  - Sequenz
    - konstruktor 176, 354
    - 2-dim. Matrixklasse 747
    - methoden (Container) 825
  - Server-Client
    - Beziehung 164
    - und callback 246
  - <set> 820, 847
  - set() 798
  - setbase() 414

- set\_difference() 733
- setf() 409
- setfill() 414
- set\_intersection() 733
- setiosflags() 414
- setprecision() 414
- setstate() 421
- set\_symmetric\_difference() 734
- set\_terminate() 338
- set\_union() 732
- setw() 414
- SFINAE 812, 970
- shared\_lock 524
- shared\_mutex 524
- shared\_ptr 373, 912
  - für Arrays 656
- SHELL 606
- shift() 920
- short 42
- showbase, showpoint, showpos 408, 415
- showContainer() 693
- shrink\_to\_fit() 827, 832, 900
- shuffle() 702
- Sichtbarkeit 62
  - sbereich (namespace) 144
  - dateiübergreifend 134
- sign() 935
- Signal 538, 541
- Signalton 56
- Signatur 122, 282, 293, 294, 296
- signed char 54
- sin(), sinh() 742, 923, 935
- single entry/ single exit 83
- size() 89, 93, 822
- size() (std::size()) 89, 207
- size\_t 48, 936
- size\_type 821
- sizeof 207
  - nicht bei dynamischen Arrays 234
- sizeof... (variadische Templates) 482
- Skalarprodukt 696
- skipws 408, 415
- sleep\_for(), sleep\_until() 502, 508
- slice 924
- slice\_array 927
- Slot 539, 541
- small string optimization 466
- Smart Pointer 367
  - und Exceptions 654
- Socket 562
- Sommerzeit 938
- Sonderzeichen 56
- sort\_heap() 738
- Sortieren
  - mit qsort() 246
  - mit sort() 714
  - stabiles 714
  - durch Verschmelzen 719
- Speicher
  - klasse 134
  - leck 221, 654
  - platzfreigabe 204
  - verwaltung (eigene) 381
- Spezialisierung
  - von Klassen 280
  - von Templates 153
- splICE() 830
- split() 672
- Sprachumgebung 876
- SQL 587
- sqrt() 52, 742, 923, 935
- <sstream> 425
- stable\_partition() 712
- stable\_sort() 715
- Stack 62
  - Klasse 271
- <stack> 787, 820, 833
- stack unwinding 331
- Standard
  - bibliothek
    - C 156, 933
    - C++ 784
  - header 157, 787
  - klassen 787
  - Typumwandlung 61, 248
  - Zeiger 248
- Standard Ein-/Ausgabe 101
- start() 924, 928
- static 134
  - Attribute und Methoden 265
  - in Funktion 113
  - Initialisierungsreihenfolge 271

- static\_assert 143
- static\_cast<>() 56, 319
- static und -Bstatic (Makefile) 618
- statisches Binden 293, 971
- Statusabfrage einer Datei 421
- std 35, 64, 145
- <stdexcept> 337
- stdio 408
- Stelligkeit (Template) 480
- stem() (Filesystem) 774
- Stichprobe 705
- STL 435
- stod() 674
- stoi() 673, 674
  - und verwandte numerische  
Konversionsfunktionen 906
- strcat(), strchr(), strcmp() 939
- strncat(), strncmp() 940
- strpbrk(), strrchr(), strstr() 940
- strcpy() 217, 939
- strncpy() 939
- stream 405
- Stream-Iterator 866
- Stream-Öffnungsarten 423
- streamsize 412
- strerror() 940
- Streuspeicherung 849
- strftime() 938
- stride() 924, 928
- String 93
  - in Zahl umwandeln 673
  - Klasse MeinString 255
  - Länge 214, 215
  - Literal 387, 880
- <string> 787, 897
- string 93, 897
  - append() 900
  - assign() 901
  - at() 93, 899
  - back() 899
  - begin() 898
  - capacity() 900
  - cbegin(), cend(), crbegin(), crend() 899
  - clear() 902
  - compare() 904
  - copy() 899
  - c\_str() 899
  - data() 899
  - empty() 899
  - end() 898
  - erase() 902
  - find() 903
  - find\_...-Methoden 904
  - front() 899
  - insert() 901
  - length() 93, 899
  - max\_size() 899
  - operator+=() 900
  - rbegin(), rend() 899
  - replace() 902
  - reserve() 900
  - resize() 900
  - shrink\_to\_fit() 900
  - size() 93, 899
  - substr() 904
  - swap() 899
- stringstream 674
- string\_view 264, 907
- <string\_view> 787
- strlen() 211, 215, 940
- strncpy() 217, 940
- strtod(), strtol(), strtoul() 937
- strtok() 940
- struct 87, 287
- Strukturierte Bindung 98
- Stunde 938
- Subobjekt 282, 289, 313
  - in virtuellen Basisklassen 315
  - verschiedene 315
- Substitutionsprinzip 309
- substr() 904
- Subtyp 283, 288, 309, 971
- Suffix 49
- sum() 920
- swap() 358, 822, 899
  - Algorithmus 761
- swap-Trick 240
- swap\_ranges() 761
- switch 72
- symmetrische Differenz (Menge) 734
- Synchronisation 510

- Syntaxdiagramm
  - ?: Bedingungsoperator 71
  - do while-Schleife 77
  - enum-Deklaration 85
  - for-Schleife 78
  - Funktions-
    - aufruf 112
    - definition 112
    - prototyp 111
    - Template 151
  - if-Anweisung 67
  - mathematischer Ausdruck 124
  - operator-Deklaration 346
  - struct-Definition 87
  - switch-Anweisung 72
  - Typumwandlungsoperator 366
  - while-Schleife 75
- system() 937
- system\_clock 502
- system\_error 337, 421
- Szenario 195
- T**
- Tabellensuche 91
- Tabulator 56
- Tag (des Monats) 938
- tan(), tanh() 742, 923, 935
- target (make) 603
- Taschenrechnersimulation 124
- Tastaturabfrage 102
- TCP 558
- »Teil-Ganzes«-Beziehung 669
- tellg(), tellp() 423
- Template
  - Alias 252
  - für Funktionen 151
  - Instanziierung von T. 273
    - explizite 624
    - ökonomische (bei vielen Dateien) 622
  - int-Parameter 274
  - für Klassen 271
  - Konstante 181
  - Metaprogrammierung 478
  - Spezialisierung 153
    - partielle 858
    - static Attribut 384
    - variable Parameterzahl 480, 748
- Template Method (Design-Muster) 307
- Template-Alias type 816
- temporäres Objekt (Vermeidung) 173
- Terminal 33
- terminate() 337
- terminate\_handler 338
- test() (Bitset) 799
- Test Driven Development 629
- Test-Suite 631
- Textersetzung 139
- this 227
- this\_thread 506
- this->, \*this bei Zugriff auf Oberklassenelement 361
- thousands\_sep() 887, 888
- Thread 501
- thread (API) 507
- Thread-Sicherheit 534
- ThreadGroup 509
- throw 332
- tie() 434
- tiefe Kopie 238
- \_\_TIME\_\_ 143
- time 884, 891
- time() 364, 939
- time\_get 891
- time\_put 892
- time\_t 364, 937
- tm 364, 937
- tolower() 679, 884, 886, 934
- top() 833, 836
- to\_string()
  - bitset 799
  - Zahlkonvertierung 906
- to\_ulong() 799
- toupper() 679, 881, 884, 886, 934
- trailing return type 129
- traits 857
- traits::eof() 412
- transform() 761, 885
- tree (Programm) 610, 779
- Trennung von Schnittstellen und Implementation 132
- true 58

- true\_type 811
- truename() 887
- trunc 423
- try 332
- Tupel, <tuple> 748, 794
- tuple\_cat() 749, 795
- Typ 971
  - polymorpher bzw. dynamischer Typ 322
- type cast *siehe* Typumwandlung
- Type Traits 810
  - Abfrage von Eigenschaften 813
  - Abfrage numerischer Eigenschaften 815
  - decay 816
  - default\_order 817
  - enable\_if 816
  - Funktionsweise 811
  - Typbeziehungen 815
  - Typumwandlungen 816
- typedef 251
- typeid() 322, 401
- type\_info 322
- <typeinfo> 337
- typename (bei Template-Parametern) 151
- Typinformation 304
  - zur Laufzeit 322
- Typumwandlung
  - cast 56, 204, 246, 319
  - durch Compiler 185
  - const\_cast<>() 321
  - dynamic\_cast<>() 320
  - mit explicit 176
  - implizit 71, 176
  - mit Informationsverlust 123
  - skonstruktoren 174, 185
  - soperator 366
    - ios 422
  - reinterpret\_cast<>() 321
  - Standard- 61
    - Zeiger 248
  - static\_cast<>() 319
- UDP 558, 567
- Überladen
  - von Funktionen 122
  - von Operatoren *siehe* operator
- Überlauf 45, 51
- Überschreiben
  - von Funktionen in abgeleiteten Klassen 292
  - virtueller Funktionen 297
- Übersetzung 130, 132
  - seinheit 132
- uintX\_t, uint\_fastX\_t, uint\_leastX\_t (X = 8, 16, 32, 64), uintmax\_t 48
- Umgebungsvariable 225
- UML 279, 663
- Umleitung der Ausgabe auf Strings 425
- unärer Operator 347
- unäres Prädikat 706
- UND
  - bitweises 46, 47
  - logisches 58
- #undef 139
- undefined behaviour 221
- underflow 52
- underflow\_error 337
- Unicode 405, 879
- uniform\_int\_distribution 754
- uniform\_real\_distribution 755
- union 99
- unique() Sequenzen 830
- unique(), -\_copy() Algorithmen 702
- unique\_lock 512
- unique\_ptr 253, 373, 466, 909
  - für Arrays 657
- Unit-Test 627
- unitbuf 408, 410, 415
- unordered\_map 853
- unordered\_multimap 855
- unordered\_multiset 856
- unordered\_set 855
- unsigned 43
- unsigned char 54
- Unterklasse 279, 971
- upper\_bound() 729, 843
- uppercase 408, 415
- URI, URL 558, 685

## U

- u16string, u32string 387
- u8 880
- UCS 880

URL-Codierung 572  
use case 194  
use\_facet() 679–681, 878  
using  
  -Deklaration 325  
  Namespace  
    Deklaration 145  
    Direktive 145  
  protected Funktion 323  
  statt typedef 251  
UTC 938  
UTF-8 879  
<utility> 278, 459, 465, 761, 790, 794

## V

Valarray  
  arithmetische Operatoren 921  
  Bit-Operatoren 922  
  logische Operatoren 922  
  mathematische Funktionen 923  
  relationale Operatoren 923  
<valarray> 917  
value(), value\_or() (optional) 809  
value\_type 821, 841, 845  
Variable 36  
  automatische 134  
  globale 134, 135  
  make 605  
Variablenname 39  
variadic templates 480, 748  
variant 800  
vector 826  
  at() 89  
  push\_back() 93  
  size() 89  
<vector> 787, 820, 826  
vector<bool> 827  
Vektor 88  
  Klasse 351  
  Länge (geom.) 697  
Verbundanweisung 66  
Vereinigung (Menge) 732  
Vererbung 279, 971  
  der abstrakt-Eigenschaft 299  
  von constraints 310  
  der Implementierung 325

  Mehrfach- 312  
  private 324  
  protected 327  
  von Zugriffsrechten 286  
  und Zuweisungsoperator 396  
Vergleich  
  von double-Werten 648  
  bei Vererbung 399  
verschmelzen (*merge*) 718  
Vertrag 310, 971  
verwitwetes Objekt 221  
Verzeichnis  
  anlegen 777  
  anzeigen 778  
  kopieren 776  
  löschen 774  
  umbenennen 777  
Verzeichnisbaum  
  anzeigen 779  
  make 611  
Verzweigung 67  
virtual 294, 296, 306  
virtuelle Basisklasse 315  
virtueller Destruktor 304  
virtuelle Funktionen 294, 296  
  private 307  
  rein- 299  
void 204  
  als Funktionstyp 111  
void\* 246  
  Typumwandlung nach 204  
  Typumwandlungsoperator 422  
volatile 972  
vollständiges Objekt 317, 972  
Vorbedingung 129, 972  
vorgegebene Parameterwerte  
  in Funktionen 121  
  in Konstruktoren 169  
Vorkommastellen 49  
Vorrangregeln 60, 945  
Vorwärtsdeklaration 199

## W

Wächter (Tabellenende) 91, 209  
Wahrheitswert 58  
  zufällig erzeugen 756

- wait() 516, 523
  - Warteschlange 834
  - wchar\_t 55, 880, 936
  - weak\_ptr 914
  - Webserver 578
  - Weite der Ausgabe 408
  - Wert
    - eines Attributs 964
    - Parameterübergabe per 115
  - Wertebereich ganzer Zahlen 44
  - Wertsemantik 239, 437, 449
    - Performanceproblem 451
  - what() 336
  - while 75, 213, 215
  - whitespace *siehe* Zwischenraumzeichen
  - wide character 55
  - widen() 886
  - Widget 541
  - width() 408
  - Wiederverwendung
    - durch Delegation 326
    - durch Vererbung 289
  - wildcard 607
  - Winterzeit 938
  - Wochentag 938
  - wofstream 883
  - Wrapperklasse für Iterator 864
  - write() 241, 407
  - ws 415
  - wstring 387, 883, 897
- X**
- XOR, bitweises 46
- Y**
- yield() 507
- Z**
- Zahl in String umwandeln 675
  - Zahlenbereich 43, 50
  - Zeichen 54
  - Zeichenkette 36, *siehe auch* String
    - C-String 210
    - Kopieren einer 215
  - Zeichenklasse (Regex) 492
  - Zeichenliteral 880
  - Zeichensatz 879
  - Zeiger 201, 217
    - Arithmetik 208
    - vs. Array 207
    - auf Basisklasse 297, 304
    - Darstellung
      - von [ ] 209
      - von [ ] [ ] 232
    - auf Elementdaten 250
    - auf Elementfunktionen 249
    - auf Funktionen 244
    - hängender 220
    - intelligente *siehe* Smart Pointer
    - Null-Zeiger 203
    - auf Oberklasse 289, 296
    - auf Objekt (Mehrfachvererbung) 314
    - auf lokale Objekte 205
    - Parameterübergabe per Z. 222
  - Zeile
    - einlesen *siehe* getline()
    - neue 56
  - Zeit-Server 570
  - Zeitkomplexität 972
  - Ziel (make) 603
  - Ziffernzeichen 54
  - Zufallszahlen 753
  - Zugriffsspezifizierer und -rechte 286
  - zusammengesetzte Datentypen 84
  - Zusicherung 142, 338
  - Zustand 972
    - eines Iterators 441
  - Zuweisung 36, 66, 70, 972
    - und Initialisierung 172
    - und Vererbung 288, 396
  - Zuweisungsoperator 172, 356, 467
    - implizite Deklaration 356, 396
    - und Vererbung 396
  - zweidimensionale Matrix 744
  - Zweierkomplement 43
  - Zwischenraumzeichen 102, 212, 411