

Stichwortverzeichnis

A

Abstrakte Klasse [227, 379](#)
Abstrakter Datentyp [379](#)
accumulate() [333](#)
Aggregat [114, 216, 380](#)
Algorithmen (C++-Bibliothek) [322](#)
Algorithmus (Definition) [16](#)
Alias-Deklaration [123](#)
Allgemeiner Konstruktor [145](#)
Anweisung [53](#)
array [114, 341](#)
ASCII [19](#)
– Tabelle [376](#)
assert [128](#)
assign() [340](#)
Assoziative Container [344](#)
at() [341](#)
– Map [348](#)
Attribute [380](#)
Aufzählungstyp enum [76](#)
Ausdruck [53](#)
Ausgabe
– -datei [82](#)
– -operator [284](#)
Ausnahme [240, 380](#)
auto [70](#)

B

back() [341, 342](#)
Basisklasse [218](#)
Bedingungsoperator [52](#)
begin() [317, 337](#)
binary_search() [326](#)
Binärdatei [86](#)
Binäre Suche [380](#)
Binärzahl [32](#)
Bit-Operatoren [34](#)
Block [22](#)
bool [49](#)
boolalpha [312](#)

C

C++11, C++14 [67](#)
C-Arrays [195](#)
case [73](#)
cerr [80](#)
char [39](#)
chrono (Zeitbibliothek) [352](#)
cin [27, 80](#)
– clear, ignore [61](#)
CircleShape (SFML) [168, 214, 253](#)
clear() [337](#)
Color (SFML) [168](#)
Compiler [17, 70](#)

complex 354
const 71
const char* 41, 192
constexpr
– Funktion 276
– Konstante 72
Container 336, 380
– -Adapter 307, 344
continue 64
copy() 328
count() (Zeitdauer) 352
cout 23, 80
CPU 387
C-String 192

D

Dateiein-/ausgabe 82
Datenkapselung 134, 381
Datentyp 22, 39, 135, 381
Datum 353
Debug 376
dec 87
= default 288, 311
#define 101
Definition 94, 382
Deklaration 94, 382
Delegation 228, 382
delete 199
deque 343
Destruktor 150
distance() 320
double 33
do-while-Schleife 59
Drei (die großen Drei) 294
duration_cast 353
dynamic_pointer_cast 226, 266

E

Editor 18
Eingabe

– -aufforderungsfenster 19
– -datei 82
– -operator 286
einheitliche Initialisierung 337
Elementfunktion 135, 219
empty() 337
end() 317, 337
endl 80
enum 76
erase() 329, 340
– Map 348
– Set 350
Ereignisgesteuerte Programmierung
164
event loop 172
Exception 240, 380
– vordefinierte 244
exception 240
explicit 141
extern 105

F

Fakultät 109
__FILE__ 242
fill()
– Algorithmus 330
– array 124
final 224
find()
– Algorithmus 324
– Map 348
– Set 350
float 33
for-Schleife 61
Formatierung der Ausgabe 88
Framework 164
friend 283
front() 341, 342
fstream 82
Funktion 91

- freie 136
- Funktionsobjekt, Funktor 310
- Fünf (die großen Fünf) 359

G

- generate() 331
- getline() 42
- Globale
 - Konstante 105
 - Variable 104
- good() 54, 61, 83, 85
- Grafik 163

H

- Header-Datei 22
- Heap (Speicherbereich) 189
- hex 87
- Hexadezimalzahl 32

I

- Identität 382
- if-Anweisung 48
- #ifdef, #ifndef 102
- ifstream 82
- Initialisierung 28, 383
- initializer_list 149, 160, 288, 338
- inline 143, 284
- inner_product() 335
- insert() 340
 - Map 348
 - Set 349
- Instanz 383
- Intervall (und -notation) 320
- iota() 331
- Iterator 317

K

- Klasse 135, 383
- Klassenvariablen und -funktionen 169

- Kommazahlen 33
- Komplexe Zahlen 354
- Konsole 19
- Konstruktor 139, 145
 - erben 220
- Kopie, flache/tiefe 291
- Kopierkonstruktor 146
 - Elimination 148
- Kopiersemantik 357

L

- Lambda-Funktion 310, 314
- Laufzeit 38
- __LINE__ 242
- Linken 18, 383
- list 342
- Liste 318, 390
- Literal 38, 384
- logic_error 244
- Logisches UND, ODER 50
- long double 33
- lower_bound() 326
- L-Wert 357, 384

M

- magic number 129, 385
- make_pair() 316
- make_shared() 205
- make_unique() 202
- Makro 22, 101
- map 345
- Mehrfachvererbung 228
- Methode 385
- min(), max() 332
- min_element(), max_element() 333
- Mitgliedsfunktion 135
- Modulo 31
- mt19937 66

N

Nachbedingung 385
Namensraum, anonymer 109
namespace 100
NDEBUG 129
Negation 52
Negationsoperator 60
Netbeans 368
new 199
now() 353
Nullbyte 43
nullptr 191

O

Oberklasse 218, 386
Objekt 135, 386
observer_ptr 272
ofstream 82
Oktalzahl 32
one definition rule 94
Open Source 386
Operator, Operand 24
operator bool() 279
operator<<() 284
operator>>() 286
operator()() 310
operator=() 291
operator==(), operator!=() 282
operator-() 284
operator->(), operator*() 273
operator+=(), operator+() 289
operator++() 279
override 223

P

pair 315
Parameterübergabe
– per Zeiger 197
– per Referenz 102
– per Wert 93

Polymorphismus 387
pop_back() 341, 342
pop_front() 342, 344
precision() 88
private: 139
protected: 218
Prototyp 93
Prozessor 387
Präprozessor 18, 22, 27, 97
public: 139
push_back() 159, 341, 342
push_front() 342, 344
put() 82, 83

R

RAII 153, 388
random_device 68
Referenz 102
– auf const 147
– -semantik 357
Reguläre Ausdrücke 365
Rekursion 106
Relationale Operatoren 49
Release 376
remove()
– Algorithmus 329
– Liste 342
resize() 341
reverse() (Liste) 342
RGB (Farben) 168
runtime_error 244
R-Wert 358, 384

S

Schnittstelle 388
Scope-Operator :: 67, 169
seed 68
set 349
setFillColor() (SFML) 168
SFML 163, 370

shared_ptr 204
Signatur 219
size() 337
sizeof 30
size_t 30
Skalarprodukt 335
sleep() 185
sort() 314, 324
Sprite 251
Stack
– Datenstruktur 307
– und Heap 189
– Laufzeit- 150
– stack (C++-Bibliothek) 344
Standardkonstruktor 145
static 170
static_assert 306
static_cast 43
std (Namespace) 23
stoi() und ähnliche 85
strcmp() 306
strcpy() 296
string 39
strlen() 193, 296
struct 216, 292
Subtyp 219, 389
swap() 337
switch 73
Syntax 18
system_clock 353
Systemfehler 245

T

Template 302
– Meta-Programmierung 363
– Spezialisierung 305
– variadisch 364
Test 366
Textur 256
this 280

Threads 365
throw 240
TicTacToe 121, 230
time_t, tm 353
to_string() 159, 242, 252
typedef 123

U

Überladen
– Funktion 95, 219
Überlauf 66
Überschreiben 219
Übersetzungseinheit 98, 389
Uhrzeit 353
uniform_int_distribution 69
uniform_real_distribution 208
unique() Sequenzen 343
unique_ptr 202
unordered_map 350
unordered_set 351
unsigned 30
Unterklasse 389
upper_bound() 327
using, Alias-Deklaration 123

V

vector 116, 157, 341
Vector2f (SFML) 174, 215, 251
Verbundanweisung 48
Vererbung 214, 389
Vergleichsoperatoren 49
Verkettete Liste 318, 390
virtual 221
void 102
Vorbedingung 389
Vorwärtsdeklaration 273

W

Wert eines Attributs 380
what() 242

while-Schleife [57](#)

whitespace [28](#)

width() [88](#)

Z

Zeiger [190](#)

Zeitmessung [352](#)

Zufallszahl (Pseudo-Z.) [66](#)

Zustand [390](#)

Zuweisung [28](#), [390](#)

Zuweisungsoperator [291](#)

Zwischenraumzeichen [28](#)