Index

A
Abrahams guarantees, 85
Abrahams, David, xiv, 45, 85
accessibility, 111
accessibility checking, 114
vs. visibility, 113, 116
Adamczyk, Steve, xiv
ADL. See Koenig lookup
Alexandrescu, Andrei, xiv, 282, 284
algorithm
spelling "algorithm", 249
algorithms
copy, 7
for_each, 7
vs. member functions, 310
Align, 283
alignment, 272
Allison, Chuck, xiv
allocation, 156
and alignment, 159, 272
and deque, 159, 200
and list, 200
and map, 159, 200
and padding, 157
and set, 160, 200
and vector, 159, 200
by compiler’s run-time library, 154
by operating system, 154
by standard containers and allocators, 154
by user-defined containers and allocators, 154
combining strategies, 153
failure recovery, 180
failures, caveats about checking, 176
fixed-size, 153
garbage-collected, 153
general-purpose, 153
housekeeping overhead, 156
lazy, 177
value of checking failures, 180
ambiguity. See disambiguation,
See declaration, vs. definition
Animal, 230
any, 281
any_cast, 281
argument-dependent lookup.
See Koenig lookup
arrays
and alignment, 159
and padding, 157
at
vector. See vector, at
Austern, Matt, xiv
auto, 218
auto_ptr
prefer shared_ptr instead of, 145
B
bad_exception, 94
BadDerived, 147
BaitAndSwitch, 107
Barfurth, Joerg, xiv
base classes
why not make concrete, 137
basic guarantee, 85
basic_string, 286
algorithms vs. member functions, 310
at, writing as nonmember function, 298
empty, writing as nonmember function, 296
erase, writing as nonmember function, 307
operator+=, writing as nonmember function, 304
resize, writing as nonmember function, 302
Becker, Pete, xiv
binary_function, 136, 262
bind2nd, 29
binders, 28
BitBuffer, 204
Boost
any, 280
Index

lexical_cast, 22, See
lexical_cast
shared_ptr, See shared_ptr
Borland, 53, 278, 283
Bray, Brandon, xiv
Budapest, ix
buffer overrun, 12
buffer overruns, 310
byte
size of in bits, 205

C
C#, 196
Calc, 110
callback, 260
callbackbase, 264
capacity
vector, See vector, capacity
Cat, 230
catch, 81, See exception safety
Caves, Jonathan, xiv
Cfront, 61, 70
checked_delete, 49
class
access to internal members, 104
class templates, 42
partial specialization, 43
prefer implementing
function templates using, 47
classes
accessibility of members, 111
code clarity, 208, 247
Colossal Cave
allusion to, 32
Comeau, 53, 59, 278, 283
Compare1stDeref, 255
CompareDeref, 256
comparing iterators, See
iterators, comparing
compression. See efficiency,
compression
const, 6
and optimization, 184, 188
and ROM, 186
avoid passing by-value
parameters as, 189
correctness, 184
const correctness, 6, 184, 251,
261, 276
cast, 6, 260
constructor
explicit, 260
implicitly generated, 145
constructors
order of execution, 100
contiguous
meaning of for allocations
and arrays, 158
conversions
floating point, 229
narrowing, 229
copy, 7
copy assignment
implicitly generated, 145
copy constructor
implicitly generated, 145
suppressing, 147
CoughtItUp, 116
Count, 139, 146
Couple, 119
cute critters, 230

D
Dalla Gasperina, Marco, 146
data compression. See
efficiency, compression
data hiding
and encapsulation, 121
declaration
vs. definition, 223
definition
vs. declaration, 223
dele, 156, See new, operator
new
dependent names, 65
deprecated
meaning of, 20
deque
example use of, 222
derivation
unsafe, 274
derived classes
enforcing base class rules,
148
design
monolithic, 286, See
monolithic design
destroy, 36
destructor
implicitly generated, 145
destructors, See RAII
and nofail guarantee, 143
public and virtual, or
protected and nonvirtual,
136
Dewhurst, Steve, xiv
digraphs, 237
Dimov, Peter, xiv, 45
Dinkumware, 279
disambiguation, 225
Disney, allusion to, 174
Dog, 230
double
  conversion to float, 229
  vs. float, 227

E
e, 194
EDG. See Edison Design Group
Edison Design Group, 53, 59, 75, 278, 283
efficiency
  and const, 184, 188
  compression, 200
  premature optimization, 193
  string formatting, 12
  using domain-specific knowledge, 210
Ellis, Margaret, 70
Employee, 27
encapsulation, 119, 122, 274, 292, 299
  breaking, 109
encrypting source code, 64
endl, 7
ends, 26
entropy, 228
errors
  error handling policy, 82
  prefer compile-time to runtime, 150
Estrada, Javier, xiv
EvaluateSalaryAndReturnName, 87
exception handling
  exception specifications, 89, 91, See exception specifications
exception safety
  and try/catch, 80
basic guarantee. See basic guarantee
  nofail guarantee. See nofail guarantee
  nothrow guarantee. See nofail guarantee
  strong guarantee. See strong guarantee
  worthwhile, 85
exception specifications, 89
  a shadow type system, 91
  and inheritance, 142
  and inline, 95
  and typedef, 92
  and virtual functions, 93
  of implicitly generated functions, 141
exceptions
  translating, 83
explicit, 260
explicit specialization, 43
export, 59, 61, 64
  and build speed, 75
  and ease of use, 74
  and Koenig lookup, 73
  and macros, 76
  and source code organization, 61
  guidelines, 76
  history, 69
  interaction with other language features, 73
  motivation, 67
  real advantages, 76
extensibility, 276, See genericity, unintentional loss of

F
FastMemory, 169
Fehér, Attila, xiv
float
  vs. double, 227
floating point. See float; double
  for_each, 7
  example use, 28
formatting
  strings, 10
Frankie, xiv
free, 156, See malloc
friends, 105, 112
  and qualification, 51
  and specializations, 51
  and templates, 49
Func, 91
function template
  specialization, 43
function templates, 42
  overloading vs. specialization, 42
  prefer implementing in terms of class templates, 47
functions
  member vs. nonmember, 292
  nontemplate are best match, 44

G
garbage collection, 153
gcc, 53, 237, 278, 283
generic programming techniques, 1
genericity
  unintentional loss of, 274, See extensibility
  unintentional loss of, 253, 259
  unintentional loss off, 36
Gnu. See gcc
Google, 237
Gregor, Doug, xiv

H
Hall, Mark, xiv
header files
  spelling of, 249
heat death, universal, 228
Heinlein, Robert, 248
Henney, Kevlin, xiv, 284
Hijack, 106
Hinnant, Howard, xiv
Horstmann, Cay, xiv
Hunc, 91
Hyslop, Jim, xiv

I
IllegalImmoralAndFattening, 270
implicit int
  and main, 249
implicitly generated functions
  suppressing, 147
inheritance
  avoid overusing, 103
inline, 190
  across languages, 195
  and exception specifications, 95
at application installation time, 195
at coding time, 192
at compile time, 193
at JIT time, 197
at link time, 194
at run time, 196
compilers better at judging benefit of, 193
function vs. function call, 193
only a hint, 193
  vs. macros, 191
_INT, 278
Intel, 53, 278, 283
invalid pointer. See deleted object
istream_iterator
  example use of, 222
Italian
  gratuitous use of, 232
iterator range
  destroy and, 36
iterators
  comparing (with !=, not <), 6
  const_iterator, 6
  operator*, 37
  pointers are, 36
J
Java, 65, 69, 73, 195, 197
Junc, 91

K
Kaminsky, Mark E., xiv
KerberosAuthentFail, 83
keywords
  auto. See auto
  list of, 217
  rationale for making reserved words, 214
  register. See register
Koenig lookup, 73
Koenig, Andrew, 179

L
Latin
  gratuitous use of, 12, 77, 175, 179, 260
lazy allocation. See allocation, lazy
length-unchecked functions never use, 18
lexical_cast, 11, 22
lifetime
  of objects, 101
lint, 13
Linux, 176
_LIST, 278

M
Machiavelli, Niccolo allusion to, 108
macros
  abuses of, 106
  and export, 76
  controlling, 76
  evils of, 232
  naming, 278
  vs. inline, 191
  vs. namespaces, 234
main
  returns int, 249
make_callback, 265
malloc, 156, See allocation
Mancl, Dennis, xiv
Manley, Kevin, 284
max munch, 239, 243
max_align, 272
McNamara, Brian, xiv
mem_fun, 27
mem_fun_ref, 28
member function pointers. See pointers, to member functions
member functions
  vs. nonmember functions, 292, 310
memory
committing backing store, 176
footprint. See allocation
physical, 176
virtual, 176, 179
memory exhaustion. See allocation
memory management, 151, See allocation
strategies, 153
memset, 178
Metrowerks, 53, 278, 283
Meyers, Scott, xiv, 169, 253, 292
Microsoft, 53, 96, 195, 278, 283
monolithic design, 259, 286
multimap
example use of, 256
Munch, Max, 239
Murphy, Capt. Edward A.
allusion to, 108
MYUNION, 273

N
name lookup, 114, 169
dependent names, 65
namespace pollution, 299
namespaces, 250
friend template in a different namespace, 50
unnamed namespace, 73
vs. macros, 234
narrowing conversions, 229
.NET, 65, 69, 130, 195
new, 156, See allocation
conformance issues, 177
in-place, 165
nothrow, 165, 174, 175
operator new. See operator, new
placement. See new, in-place
No, 111
Satisfaction, 111
nonmember functions
vs. member functions, 292, 310
Nonvirtual Interface pattern, 132
nothrow guarantee. See nofail guarantee
numeric_limits
example use of, 206
NVI. See Nonvirtual Interface pattern
O
object layout
abusing, 107
object lifetime, 41, 101
ODR, 61
and export, 74
One Definition Rule. See ODR operator
(), 261
->, prefer prefix form, 6
++, prefer prefix form, 6
=, 143
new, 156, 165, See allocation
new, and name hiding, 168
new, class-specific, 167
new, replacing, 166, 170
operator=. See --
operator new. See operator, new
operator[]. See vector, operator[]
operator++. See ++
operators
&, 37, 241
&&, 241
*, 37
?, 243
|, 242
||, 242
+, 240
++, 240
+=, 304
<, 242
<<, 242
optimization. See efficiency, See efficiency
order of construction, 100
Orwell, George
Animal Farm, 10
out_of_range, 3
overload resolution, 114
and export, 73
overloading
function templates and, 44
of primary function templates, 44
prefer to specialization, 47
specializations not considered, 44
vs. specialization. See function templates, overloading vs. specialization
P
pair, 126
Pair2nd, 256
parameters
avoid passing by const value, 189
partial specialization, 43
pathological code, 178
peekaboo parameters. See standard library, signatures of member functions
Peil, Jeff, xiv
persecuted pariahs
stream, 20
perversity, 158
pi, 194
Pimpl idiom, 63
plain old data. See POD
Plauger, P.J., xiv, 249
POD, 178
pointer
  invalid. See deleted object
to function, 92
pointers
  iterators aren’t always, 36
to member functions, 29
policy-based design, 259
polymorphism
  and encapsulation, 121
  combining static and
dynamic, 263
  compile-time vs. run-time, 31
Potter, John, xiv
preincrement
  prefer to postincrement, 253
premature optimization
  se. See efficiency, premature
optimization
PrettyFormat, 10
primary template, 43
protected data
  why evil, 123
public data
  why evil, 123
public member functions
  prefer to make nonvirtual, 129
puppy. See Frankie

S
Sebor, Martin, xiv
security, 310
security issues. See buffer
  overrun
semicolon
  at end of for statement, 236
  separate compilation
    of functions, 63
    of templates. See export
Session, 83
shared_ptr
  example use of, 264
  prefer instead of auto_ptr, 145
size
  vector. See vector, size
Slaughter, James, xiv
Sleep
  and SleepEx, 230
SleepEx, 230
Smirnov, Nikolai, xiv
sprintf, 11, 16
  nonstandard variants, 17
  _snprintf, 17
Socrates, xi
Socratic method, xi
sort_idxtbl, 246
sort_idxtbl_pair, 246
source code
  encrypting, 64
specialization
  and friends, 51
  befriending in a different
    namespace, 55
  explicit, 43
  of class template, 43
  of function template, 43
  partial, 43
  prefer overloading instead,
    47
  vs. overloading. See function
templates, overloading vs.
specialization
specializations
  don’t overload, 46
Spicer, John, xiv, 75, 91
sprintf, 10
  never use, 18
Square, 191
Stack, 87
standard library
  implementation latitude, 30
  signatures of member
defines, 29
Star Trek
  gratuitous reference to, 230
Starbucks
  gratuitous advertisement for, 129
Stepanov, Alexander, 71
string. See basic_string
  string formatting, 10
  _STRING, 278
stringstream, 11, 18
strong guarantee, 85
Stroustrup, Bjarne, xiv, 13, 70, 116, 123, 205
strstream, 11, 20
subtle mistakes
printf formatting strings, 14
swap, 38, 40
exception safety, 38
specializing or overloading for your own types, 39, 40

T
tails
twisting compilers', 55
templatability, 14
template, primary, 43
templates
advantages of befriending with <...> syntax, 58
and friends, 49
and source code
organization, 61
cascading recompilation, 66
class. See class templates
dependent names, 65
export. See export
function. See function
templates
history, 70
inclusion model, 59, 60
member templates and access abuses, 108
separation model, 61
separation model (export), 59
source code exposure, 62
source dependencies, 62
Turing-complete, 70
terminate, 94
tests
unit tests, 206
throw, 81, 83, 179
Tina, v, xiv
tokenizing. See max munch
transform
example use of, 256
trigraphs
evils and surprises of, 236
try, 80, See exception safety
Turing-completeness
and templates, 70
Twice, 110
type safety, 13
typename
example uses of, 266

U
unary_function, 136, 261
unexpected, 90, 94
unions, 269
discriminated, 282
uniontype, 278
unit tests, 206

V
van Winkel, Jan Christiaan, xiv, 100
Vandevoorde, Daveed, xiv
Variant, 282
vector, 2
at, 3
capacity, 4
iterator not necessarily a pointer, 6
operator[], 3
reserve, 4
resize, 5
size, 5
vector<bool> specialization, 208
versioning
and robustness, 133
vexing parse. See declaration, vs. definition
virtual functions
calling nonvirtually, 197
prefer to make nonpublic, 129
visibility
vs. accessibility, 113, 116

W
Wade, Bill, xiv
Webster, Noah, 120
Weevil, 230
whitespace. See auto; register
width specifiers
minimum vs. maximum, 12
WinFX Design Guidelines, 131

X
Xyzzy, 32