Index

**Bold face** numbers refer to the page on which the term was first defined.
Numbers in *italic* refer to the “Defined Terms” section in which the term is defined.

... (ellipsis parameter), 244
/* */ (block comment), 10, 28
// (single-line comment), 10, 28
_DATE_, 221
_FILE_, 221
_LINE_, 221
_TIME_, 221
_cppplusplus, 803
\0 (null character), 40
\Xnnn (hexadecimal escape sequence), 40
\n (newline character), 40
\t (tab character), 40
{} (curly brace), 3, 28
#define, 71
#undef, 71
#include, 7
~classname, see destructor
L’c’ (wchar_t literal), 40
crlf (Unix end-of-file), 20
crlf (Windows end-of-file), 20
; (semicolon), 3
    class definition, 440
++ (increment), 13, 30, 146, 190
    and dereference, 163
    iterator, 98, 108, 312
    overloaded operator, 527
    pointer, 114
    prefix yields lvalue, 162
    reverse iterator, 412
-- (decrement), 190
    and dereference, 163
    iterator, 312
    overloaded operator, 527
    prefix yields lvalue, 162
    reverse iterator, 412
* (dereference), 98, 108, 146, 389
    iterator, 98
    on map yields pair, 362
    overloaded operator, 524
    pointer, 119
    yields lvalue, 99, 120
& (address-of), 115, 146, 511
- > (arrow operator), 164
    class member access, 445
    overloaded operator, 525
- >* (pointer to member arrow), 783
[ ] (subscript), 87, 108, 146
    bitset, 105
deque, 325
map, 363
string, 87
vector, 94, 325
    and multi-dimensional array, 142
    and pointer, 124
array, 113
    overloaded operator, 522
    reference return, 522
valid subscript range, 88
    yields lvalue, 88
() (call operator), 25, 30, 226, 280
    overloaded operator, 530
:: (scope operator), 8, 30, 78, 108
    base class members, 569
    class member, 85, 445
    container defined type, 317
    member function definition, 262
    to override name lookup, 449
= (assignment), 13, 30, 159
    and conversion, 179
    and equality, 161
    class assignment operator, 476
    container, 328
    overloaded operator, 483, 520
    and copy constructor, 484
    check for self-assignment, 490
Message, 490
multiple inheritance, 737
reference return, 483, 521
rule of three, 485
use counting, 495, 498
valuelike classes, 501
pointer, 120
string, 86
to signed, 37
to unsigned, 37
yields lvalue, 160
+= (compound assignment), 13, 30, 161
string, 86
iterator, 313
overloaded operator, 511
Sales_item, 521
+ (addition), 150
string, 86
iterator, 101, 313
pointer, 123
Sales_item, 517
- (subtraction), 150
iterator, 101, 313
pointer, 123
* (multiplication), 150
string, 85
algorithm, 421
container, 322
container adaptor, 350
iterator, 98, 312
string, 347
!= (inequality), 30, 154
container, 322
container adaptor, 350
iterator, 98, 312
string, 347
<= (less-than), 30, 153
overloaded and containers, 520
used by algorithm, 420
<= (less-than-or-equal), 13, 30, 153
> (greater-than), 30, 153
>= (greater-than-or-equal), 30, 153
>> (input operator), 8, 30
Sales_item, 516
istream_iterator, 408
string, 81, 108
overloaded operator, 515
must be nonmember, 514
precedence and associativity, 158
<< (output operator), 7, 30
bitset, 106
ostream_iterator, 408
string, 81, 108
formatting, 514
overloaded operator, 513
must be nonmember, 514
precedence and associativity, 158
Sales_item, 514
>> (right-shift), 155, 190
<< (left-shift), 155, 190
& & (logical AND), 152
operand order of evaluation, 172
overloaded operator, 511
| | (logical OR), 152
operand order of evaluation, 172
overloaded operator, 511
& (bitwise AND), 156, 189
Query, 610
! (logical NOT), 153
| (bitwise OR), 156, 190
equivalent, 290
Query, 610
^ (bitwise XOR), 156, 190
~ (bitwise NOT), 156, 190
, (comma operator), 168, 189
example, 289
operand order of evaluation, 172
overloaded operator, 511
?: (conditional operator), 165, 189
operand order of evaluation, 172
+ (unary plus), 150
- (unary minus), 150
\ nnn (octal escape sequence), 40
ddd.dddd or ddd.ddddd (long double literal), 39
numEnum or numenum (double literal), 39
numF or numf (float literal), 39
numL or numl (long literal), 39
numU or numa (unsigned literal), 39
class member : constant expression, see bit-field

A
abnormal termination, stream buffers, 292
abort, 692, 748
absInt, 530
### Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>abstract base class, 596, 621</td>
<td></td>
</tr>
<tr>
<td>example, 609</td>
<td></td>
</tr>
<tr>
<td>abstract data type, 78, 429, 473</td>
<td></td>
</tr>
<tr>
<td>abstraction, data, 432, 474</td>
<td></td>
</tr>
<tr>
<td>access control, 65</td>
<td></td>
</tr>
<tr>
<td>in base and derived classes, 570</td>
<td></td>
</tr>
<tr>
<td>local class, 796</td>
<td></td>
</tr>
<tr>
<td>nested class, 787</td>
<td></td>
</tr>
<tr>
<td>using declarations to adjust, 573</td>
<td></td>
</tr>
<tr>
<td>access label, 65, 609, 622, 628, 65, 432</td>
<td></td>
</tr>
<tr>
<td>private, 65, 432</td>
<td></td>
</tr>
<tr>
<td>protected, 562, 622</td>
<td></td>
</tr>
<tr>
<td>public, 65, 432</td>
<td></td>
</tr>
<tr>
<td>Account, 468</td>
<td></td>
</tr>
<tr>
<td>accumulate, 396, 823</td>
<td></td>
</tr>
<tr>
<td>Action, 783</td>
<td></td>
</tr>
<tr>
<td>adaptor, 348, 353</td>
<td></td>
</tr>
<tr>
<td>container, 348</td>
<td></td>
</tr>
<tr>
<td>function, 533, 535, 553</td>
<td></td>
</tr>
<tr>
<td>iterator, 399</td>
<td></td>
</tr>
<tr>
<td>addition (+), 150</td>
<td></td>
</tr>
<tr>
<td>string, 86</td>
<td></td>
</tr>
<tr>
<td>iterator, 101, 313</td>
<td></td>
</tr>
<tr>
<td>pointer, 123</td>
<td></td>
</tr>
<tr>
<td>Sales_item, 517</td>
<td></td>
</tr>
<tr>
<td>address, 35, 73</td>
<td></td>
</tr>
<tr>
<td>address-of (&amp;), 115, 146</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 511</td>
<td></td>
</tr>
<tr>
<td>adjacent_difference, 824</td>
<td></td>
</tr>
<tr>
<td>adjacent_find, 813</td>
<td></td>
</tr>
<tr>
<td>algorithm, 392, 424</td>
<td></td>
</tr>
<tr>
<td>_copy versions, 400, 421</td>
<td></td>
</tr>
<tr>
<td>_if versions, 421</td>
<td></td>
</tr>
<tr>
<td>element type constraints, 394</td>
<td></td>
</tr>
<tr>
<td>independent of container, 393</td>
<td></td>
</tr>
<tr>
<td>iterator argument constraints, 397, 415</td>
<td></td>
</tr>
<tr>
<td>iterator category and, 416, 418</td>
<td></td>
</tr>
<tr>
<td>naming convention, 420–421</td>
<td></td>
</tr>
<tr>
<td>overloaded versions, 420</td>
<td></td>
</tr>
<tr>
<td>parameter pattern, 419–420</td>
<td></td>
</tr>
<tr>
<td>passing comparison function, 403</td>
<td></td>
</tr>
<tr>
<td>read-only, 396</td>
<td></td>
</tr>
<tr>
<td>structure, 419</td>
<td></td>
</tr>
<tr>
<td>that reorders elements, 421</td>
<td></td>
</tr>
<tr>
<td>that writes elements, 398</td>
<td></td>
</tr>
<tr>
<td>type independence, 394, 396</td>
<td></td>
</tr>
<tr>
<td>using function object as argument, 531</td>
<td></td>
</tr>
<tr>
<td>with two input ranges, 420</td>
<td></td>
</tr>
<tr>
<td>algorithm header, 395</td>
<td></td>
</tr>
<tr>
<td>alias, namespace, 720, 750</td>
<td></td>
</tr>
<tr>
<td>allocator, 755, 755–759, 805</td>
<td></td>
</tr>
<tr>
<td>allocate, 759</td>
<td></td>
</tr>
<tr>
<td>compared to operator new, 761</td>
<td></td>
</tr>
<tr>
<td>construct, 755, 758</td>
<td></td>
</tr>
<tr>
<td>compared to placement new, 762</td>
<td></td>
</tr>
<tr>
<td>deallocate, 759</td>
<td></td>
</tr>
<tr>
<td>compared to operator delete, 761</td>
<td></td>
</tr>
<tr>
<td>destroy, 755, 759</td>
<td></td>
</tr>
<tr>
<td>compared to calling destructor, 763</td>
<td></td>
</tr>
<tr>
<td>operations, 756</td>
<td></td>
</tr>
<tr>
<td>alternative operator name, 46</td>
<td></td>
</tr>
<tr>
<td>ambiguous</td>
<td></td>
</tr>
<tr>
<td>conversion, 541–544</td>
<td></td>
</tr>
<tr>
<td>multiple inheritance, 734</td>
<td></td>
</tr>
<tr>
<td>function call, 269, 272, 280</td>
<td></td>
</tr>
<tr>
<td>multiple base classes, 738</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 550</td>
<td></td>
</tr>
<tr>
<td>AndQuery, 609</td>
<td></td>
</tr>
<tr>
<td>definition, 618</td>
<td></td>
</tr>
<tr>
<td>eval function, 619</td>
<td></td>
</tr>
<tr>
<td>anonymous union, 795, 805</td>
<td></td>
</tr>
<tr>
<td>app (file mode), 296</td>
<td></td>
</tr>
<tr>
<td>append, string, 342</td>
<td></td>
</tr>
<tr>
<td>argc, 244</td>
<td></td>
</tr>
<tr>
<td>argument, 25, 28, 226, 227, 280</td>
<td></td>
</tr>
<tr>
<td>array type, 238</td>
<td></td>
</tr>
<tr>
<td>C-style string, 242</td>
<td></td>
</tr>
<tr>
<td>const reference type, 235</td>
<td></td>
</tr>
<tr>
<td>conversion, 229</td>
<td></td>
</tr>
<tr>
<td>with class type conversion, 541</td>
<td></td>
</tr>
<tr>
<td>copied, 230</td>
<td></td>
</tr>
<tr>
<td>uses copy constructor, 478</td>
<td></td>
</tr>
<tr>
<td>default, 253</td>
<td></td>
</tr>
<tr>
<td>iterator, 238, 242</td>
<td></td>
</tr>
<tr>
<td>multi-dimensioned array, 241</td>
<td></td>
</tr>
<tr>
<td>passing, 229</td>
<td></td>
</tr>
<tr>
<td>pointer to const, 231</td>
<td></td>
</tr>
<tr>
<td>pointer to nonconst, 231</td>
<td></td>
</tr>
<tr>
<td>reference parameter, 233</td>
<td></td>
</tr>
<tr>
<td>template, see template argument</td>
<td></td>
</tr>
<tr>
<td>to main, 243</td>
<td></td>
</tr>
<tr>
<td>to member function, 260</td>
<td></td>
</tr>
<tr>
<td>nonconst reference parameter, 235</td>
<td></td>
</tr>
<tr>
<td>type checking, 229</td>
<td></td>
</tr>
<tr>
<td>ellipsis, 244</td>
<td></td>
</tr>
<tr>
<td>of array type, 239</td>
<td></td>
</tr>
<tr>
<td>of reference to array, 240</td>
<td></td>
</tr>
<tr>
<td>with class type conversion, 541</td>
<td></td>
</tr>
<tr>
<td>argument deduction, template, 637</td>
<td></td>
</tr>
<tr>
<td>argument list, 226</td>
<td></td>
</tr>
</tbody>
</table>
argv, 244
arithmetic
    iterator, 100, 107, 312, 313
    pointer, 123, 146
arithmetic operator
    and compound assignment, 162
    function object, 533
    overloaded operator, 517
arithmetic type, 34, 73
    conversion, 180, 188
        from bool, 182
        signed to unsigned, 36
    conversion to bool, 182
array, 40, 73, 110–114
    and pointer, 122
    argument, 238
    as initializer of vector, 140
    assignment, 112
    associative, 388
    conversion to pointer, 122, 181
        and template argument, 639
    copy, 112
    default initialization, 111
        uses copy constructor, 478
        uses default constructor, 460
    definition, 110
    elements and destructor, 485
    function returning, 228
    initialization, 111
    multi-dimensioned, 141–144
        and pointer, 143
    definition, 142
        initialization, 142
    parameter, 241
    subscript operator, 142
    of char initialization, 112
    parameter, 238–244
    buffer overflow, 242
    convention, 241–243
    reference type, 240
    size calculation, 309
    and sizeof, 167
    subscript operator, 113
arrow operator (->), 164
    auto_ptr, 704
    class member access, 445
    generic handle, 670
    overloaded operator, 525
assert preprocessor macro, 221, 223
assign
    container, 328
string, 340
assignment
    vs. initialization, 49
    memberwise, 483, 503
assignment (=), 13, 30, 159, 502
    and conversion, 179
    and copy constructor, 484
    check for self-assignment, 490
    container, 328
    for derived class, 586
    Message, 490
    multiple inheritance, 737
    overloaded operator, 476, 483, 520
    reference return, 483, 521
    pointer, 120
    rule of three, 485
        exception for virtual destructors, 588
string, 86
    synthesized, 483, 503
    to base from derived, 578
    to signed, 37
    to unsigned, 37
    use counting, 495, 498
    usually not virtual, 588
    valuelike classes, 501
    yields lvalue, 160
associative array, see map, 388
associative container, 356, 388
    assignment (=), 328
    begin, 369
    clear, 359
    constructors, 360
    count, 377
    element type constraints, 309, 323
    empty, 359
    equal_range, 379
    erase, 359
    find, 377
    insert, 364
    key type constraints, 360
    lower_bound, 377
    operations, 358
    overriding the default comparison, 604
    rbegin, 412
    rend, 412
    returning an, 381
    reverse_iterator, 412
    size, 359
    supports relational operators, 359
swap, 329
  types defined by, 361
upper_bound, 377
associativity, 149, 170, 188
  overloaded operator, 507
at
ddeque, 325
  vector, 325
ate (file mode), 296
auto_ptr, 702, 748
  constructor, 703
  copy and assignment, 704
  default constructor, 705
  get member, 705
  operations, 703
  pitfalls, 707
  reset member, 706
  self-assignment, 705
automatic object, 255, 280
  see also local variable
  see also parameter
  and destructor, 485

B
back
dqueue, 352
  sequential container, 324
dback_inserter, 399, 406, 424
dbad, 289
dbad_alloc, 175, 219
  bad_cast, 219, 774
  bad_typeid, 776
badbit, 289
base, 414
base class, 285, 302, 558, 621
  see also virtual function
  abstract, 596, 621
  example, 609
  access control, 561, 571
  assignment operator, usually not virtual, 588
  can be a derived class, 566
  constructor, 581
    calls virtual function, 589
    not virtual, 588
  conversion from derived, 567
    access control, 579
  definition, 560
  destructor
    calls virtual function, 589
    usually virtual, 587
    friendship not inherited, 575
    handle class, 599
  member operator delete, 764
  member hidden by derived, 593
  multiple, see multiple base class
  must be complete type, 566
  no conversion to derived, 580
  object initialized or assigned from derived, 578
  scope, 590
  static members, 576
  user, 563
  virtual, 751
    see virtual base class
Basket, 605
total function, 606
Bear, 731
  as virtual base, 741
begin, 353
  map, 369
  set, 372
  vector, 97
  container, 317
  best match, 269, 280
    see also function matching
    see also function matching
bidirectional iterator, 417, 424
  list, 417
  map, 417
  set, 417
  binary (file mode), 296
  binary function object, 533
  binary operator, 148, 188
  binary_search, 814
BinaryQuery, 609
definition, 617
bind1st, 535
bind2nd, 535
binder, 535, 552
binding, dynamic, 559, 621
  requirements for, 566
bit-field, 798, 805
  access to, 798
bitset, 101, 101–106, 107
  any, 104
  count, 104
  flip, 105
    compared to bitwise NOT, 155
  none, 104
  reset, 105
  set, 105
size, 104
test, 105
to_ulong, 105
compared to bitwise operator, 156
constructor, 101–103
header, 101
output operator, 106
subscript operator, 105
bitwise AND (&), 156, 189
example, 610
bitwise exclusive or (\^), 156, 190
bitwise NOT (~), 155, 189
example, 610
bitwise operator, 154–159
and compound assignment, 162
compared to bitset, 156
compound assignment, 157
example, 290
operand, 155
bitwise OR (|), 156, 190
example, 290, 610
block, 3, 13, 28, 193, 223
try, 216, 217, 224, 750
as target of if, 196
function, 227
block scope, 193
body, function, 3, 29, 226, 281
book finding program
using equal_range, 379
using find, 377
using upper_bound, 378
bookstore program, 26
exception classes, 698
bool, 35
and equality operator, 154
conversion to arithmetic type, 182
literal, 39
bool alpha manipulator, 826
brace, curly, 3, 28
break statement, 212, 223
and switch, 201–203
buffer, 8, 28
flushing, 290
buffer overflow, 114
and C-style string, 132
array parameter, 242
built-in type, 3, 28, 34–37
class member default initialization, 264
classification, 179
initialization of, 51
Bulk_item
class definition, 564
constructor, 581
constructor using default arguments, 582
derived from Disc_item, 584
interface, 558
member functions, 559
byte, 35, 73

C

calling C function from C++, 801
compiling C and C++, 803
using C linkage, 802
.C file, 4
.cc file, 4
.cpp file, 4
.cp file, 4
C library header, 90
C with classes, 430
C-style cast, 186
C-style string, 112, 130, 130–134, 145
and char*, 131
and string literal, 140
called to string, 134, 138
definition, 130
dynamically allocated, 138
initialization, 130
parameter, 242
pitfalls with generic programs, 671
c_str, 140
example, 294
CachedObj
add_to_freelist, 771
operator delete, 770
operator new, 769
allocation explained, 769
definition, 767
definition of static members, 771
design, 766
illustration, 767
inheriting from, 768
call operator (\()\), 25, 30, 226, 280
exclusion flow, 226
overloaded operator, 530
calling C function from C++, 801
candidate function, 270, 280
and function templates, 679
namespaces, 728
overloaded operator, 549

capacity
  string, 336
  vector, 331
case label, 201, 201–204, 223
default, 203
cassert header, 221
cast, 183, 188
  checked, see dynamic_cast
  old-style, 186
catch clause, 216, 217, 223, 693, 749
  catch(...), 696, 748
  example, 217
  exception specifier, 693
  matching, 693
  ordering of, 694
  parameter, 693
category, iterator, 425
cctype, 88–89, 107
  header, 88
cerr, 6, 28
char, 34
  literal, 40
char string literal, see string literal character
  newline (\n), 40
  nonprintable, 40, 75
  null (\0), 40
  printable, 88
  tab (\t), 40
checked cast, see dynamic_cast
CheckedPtr, 526
children’s story program, 400
revisited, 531
cin, 6, 28
  by default tied to cout, 292
c1, 4
class, 20, 28, 63, 73, 473
  static member, 64, 74
  const vs. mutable, 443
  const, initialization, 455
  constraints on type, 438
definition, 435
  initialization, 454
  mutable, 443
  reference, initialization, 455
  static, 469
data member definition, 65
default access label, 433
default inheritance access label, 574
definition, 64, 430–440
  and header, 264, 437
derived, 285, 302, see derived class, 621
destructor definition, 486
direct base, see immediate base class, 621
explicit constructor, 462
forward declaration, 438
generic handle, 667, 683
handle, see handle class, 599, 622
  immediate base, 566, 622
  indirect base, 566, 622
local, see local class, 806
member, 64, 73, 430
member access, 445
member function, 25, 29, see member function
member: constant expression, see bitfield
  multiple inheritance, see multiple base class
nested, see nested class, 806
nonvirtual function, calls resolved at compile time, 569
operator delete, see member operator
operator new, see member operator
pointer member
  copy control, 492–501
  copy control strategies, 499
default copy behavior, 493
pointer to member, 780
  definition, 781
pointer to member function, definition, 782
preventing copies, 481
private member, 431
Index

inheritance, 561
private member, 75
protected member, 562
public member, 75, 431
inheritance, 561
static member, 467
as default argument, 471
data member as constant expression, 471
element, 468
inheritance, 576
template member, see member template
type member, 435
undefined member, 482
user, 433, 563
virtual base, 751
virtual function, calls resolved at run time, 569
class, keyword, 64
compared to typename, 631
in template parameter, 630
in variable definition, 440
class keyword, 473
class declaration, 438, 473
of derived class, 566
class derivation list, 563, 621
access control, 570
default access label, 574
multiple base classes, 731
virtual base, 742
class scope, 65, 444, 473
friend declaration, 466
inheritance, 590–595
member definition, 445
name lookup, 447
static members, 470
virtual functions, 594
class template, 90, 107, 627, 683
see also template parameter
see also template argument
see also instantiation
compiler error detection, 634
declaration, 629
definition, 627
error detection, 635
explicit template argument, 636
export, 645
friend
declaration dependencies, 658
explicit template instantiation, 657
non-template class or function, 656
template class or function, 656
member function, 653
defined outside class body, 651
instantiation, 653
member specialization, 677
member template, see member template
nontype template parameter, 655
static member, 665
accessed through an instantiation, 666
definition, 666
type includes template argument(s), 628, 637
type-dependent code, 634
uses of template parameter, 649
class template specialization
definition, 675
member, declaration, 677
member defined outside class body, 676
namespaces, 730
class type, 20, 28, 65
class member default initialization, 264
conversion, 183
initialization of, 52
object definition, 439
parameter and overloaded operator, 507
variable vs. function declaration, 460
cleanup, object, see destructor
clear, 289, 290
associative container, 359
element, 290, 295
sequential container, 327
clog, 6, 28
close, 294
comma operator (, ), 168, 189
example, 289
operand order of evaluation, 172
overloaded operator, 511
comment, 10, 28
block (/* */), 10, 28
single-line (//), 10, 28
compare
plain function, 624
string, 347
template version, 625
instantiated with pointer, 671
specialization, 672
compilation
  and header, 67
  conditional, 220
inclusion model for templates, 644
needed when class changes, 434
needed when inline function changes,
  258
separate, 67, 76
  of templates, 643
separate model for templates, 644
compiler
  extension, 112
  flag for inclusion compilation model,
    645
  GNU, 4
  Microsoft, 4
  template errors diagnosed at link time,
    635
compiler extension, 145
compiling C and C++, 803
composition vs. inheritance, 573
compound assignment (e.g., +=), 13,
  30, 161
  string, 86
    bitwise operator, 157
    iterator, 313
    overloaded operator, 511, 518
Sales_item, 521
compound expression, 168, 188
compound statement, 193, 223
compound type, 58, 73, 145
compute, 542
  overloaded version, 545
concatenation
  Screen operations, 441
  string, 86
  string literal, 41
concrete class, 433
  initialization, 464
condition, 13, 28
  and conversion, 179
    assignment in, 161
    in do while statement, 211
    in for statement, 15, 207
    in if statement, 18, 195
    in logical operator, 152
    in while statement, 205
    stream type as, 19, 183, 288
    string input operation as, 82
condition state, 287, 302
conditional compilation, 220
conditional operator (?:), 165, 189
  operand order of evaluation, 172
console window, 6
const, 57
  and dynamically allocated array, 136
    conversion to, 182, 231
    and template argument, 639
    iterator vs. const_iterator, 100
    object scope, 57, 69
    overloading and, 267, 275
    parameter, 231
    pointer, 128
    reference, 59
    initialization, 60
const object, constructor, 453
const data member
  static data member, 470
  compared to mutable, 443
  initialization, 455
const member function, 261, 262, 280,
  431, 473
  overloaded, 442
  reference return, 442
  static, 469
const pointer, see also pointer to const
  conversion from nonconst, 182
const reference
  argument, 235
    conversion from nonconst, 182
    parameter, 235
    overloading, 275
    return type, 249
const void*, 127, 145
const_cast, 183, 184
const_iterator, 99, 415
  compared to const_iterator, 100
    container, 316
const_reference, 317
const_reverse_iterator, 412
    container, 316
constant expression, 62, 74
  and header file, 69
    array index, 110
    bit-field, 798
    enumerator, 62
    nontype template parameter, 633
    static data member, 471
construction, order of, 456, 749
  derived objects, 581, 582
  multiple base classes, 732
  virtual base classes, 746
constructor, 49, 74, 262, 281, 431
const objects, 453
conversion, 461, 474
  function matching, 547
  with standard conversion, 540
copy, 476–482, 502
  base from derived, 578
multiple inheritance, 737
default, 52, 74, 262, 281, 458–461, 474
default argument in, 458
derived class, 581
  initializes immediate base class, 583
  initializes virtual base, 744
execution flow, 454
explicit, 462, 474
  copy-initialization, 477
for associative container, 360
for sequential container, 307–309
function try block, 696
in constructor initializer list, 457
inheritance, 581
initializer, 452
may not be virtual, 588
object creation, 452
order of construction, 456
derived objects, 581, 582
multiple base classes, 732
  virtual base classes, 746
overloaded, 452
pair, 357
resource allocation, 700
synthesized copy, 479, 503
synthesized default, 264, 281, 459, 474
virtual inheritance, 744
constructor initializer list, 263, 281, 431,
  453–458, 474
  compared to assignment, 454
  derived classes, 582
  function try block, 696
  initializers, 457
  multiple base classes, 733
  sometimes required, 455
  virtual base class, 745
container, 90, 107, 306, 353
  see also sequential container
  see also associative container
  and generic algorithms, 393
as element type, 311
assignment (=), 328
associative, 356, 388
begin, 317
clear, 327
const_iterator, 316
const_reference, 317
const_reverse_iterator, 316
element type constraints, 309, 323
elements and destructor, 485
elements are copies, 318
empty, 323
end, 317
erase, 402
has bidirectional iterator, 417
inheritance, 597
insert, 319
iterator, 316
rbegin, 317, 412
reference, 317
rend, 317, 412
returning a, 381
reverse_iterator, 316, 412
sequential, 306, 354
size, 323
size_type, 316
supports relational operators, 321
swap, 329
types defined by, 316
continue statement, 214, 223
  example, 290
control, flow of, 192, 224
conversion, 178, 188
  ambiguous, 541–544
  and assignment, 159
  argument, 229
    with class type conversion, 541
    arithmetic type, 180, 188
  array to pointer, 122, 238
    and template argument, 639
class type, 183, 535, 552
  design considerations, 544
    example, 537
  multiple conversions lead to am-
    biguities, 546
  operator, 337, 537–540, 552
  operator and function matching, 545
    used implicitly, 538
    with standard conversion, 539
compiler, 461
  function matching, 547
    with standard conversion, 540
derived to base, 567, 580
  access control, 579
enumeration type to integer, 182
from istream, 183
function matching of template and non-templated functions, 681
function to pointer, 277
and template argument, 639
implicit, 189
inheritance, 577
integral promotion, 180
multi-dimensioned array to pointer, 143
multiple inheritance, 734
non-templated type argument, 640
of return value, 246
rank for function matching, 272
rank of class type conversions, 545
signed type, 180
signed to unsigned, 36
template argument, 638
to const pointer, 127
to const, 182
and template argument, 639
parameter matching, 231
virtual base, 743
conversion constructor, 474
copy, 815
copy constructor, 476, 476–482, 502
and assignment operator, 484
argument passing, 478
base from derived, 578
for derived class, 586
initialization, 478
Message, 489
parameter, 480
pointer members, 480
rule of three, 485
exception for virtual destructors, 588
synthesized, 479, 503
use counting, 495, 497
value-like classes, 500
copy control, 476, 502
handle class, 601
inheritance, 584–590
message handling example, 489
multiple inheritance, 737
of pointer members, 499
copy-initialization, 48
using constructor, 477
copy_backward, 816
count, use, 495, 503
count, 812
book finding program, 377
map, 367
multimap, 377
multiset, 377
set, 372
count_if, 404, 812
with function object argument, 532
cout, 6, 28
by default tied to cin, 292
cstddef header, 104, 123
cstdlib header, 247
cstring header, 132
curly brace, 3, 28
D
dangling else, 198, 223
dangling pointer, 176, 188
returning pointer to local variable, 249
synthesized copy control, 494
data abstraction, 432, 474
advantages, 434
data hiding, 434
data structure, 20, 28
data type, abstract, 473
dec manipulator, 827
decimal literal, 38
declaration, 52, 74
class, 438, 473
class template member specialization, 677
dependencies and template friends, 658
derived class, 566
export, 645
forward, 438, 474
function, 251
exception specification, 708
function template specialization, 672, 673
member template, 661
template, 629
using, 78, 108, 720, 750
access control, 573
class member access, 574
overloaded inherited functions, 593
declaration statement, 193, 224
decrement (–), 190
overloaded operator, 526
prefix yields lvalue, 162
reverse iterator, 412
deduction, template argument, 637
default argument, 253
and header file, 254
function matching, 270
in constructor, 458
initializer, 254
overloaded function, 267
virtual functions, 570
default case label, 203, 224
default constructor, 52, 74, 262, 281, 458–461, 474
Sales_item, 263
string, 52, 81
default argument, 458
synthesized, 264, 281, 459, 474
used implicitly, 459
variable definition, 460
definition, 52, 74
array, 110
base class, 560
C-style string, 130
class, 64, 430–440
class data member, 65, 435
class static member, 469
class template, 627
   static member, 666
class template specialization, 675
   member defined outside class body, 676
class type object, 439
derived class, 563
destructor, 486
dynamically allocated array, 135
dynamically allocated object, 174
function, 3
inside an if condition, 196
inside a switch expression, 203
inside a while condition, 205
map, 360, 373
multi-dimensioned array, 142
namespace, 712
can be discontiguous, 714
member, 716
of variable after case label, 204
overloaded operator, 482
pair, 356
pointer, 115
pointer to function, 276
static data member, 470
variable, 48
delete, 145, 176, 188, 806
   compared to operator delete, 760
const object, 178
execution flow, 760
member operator, 806
member operator
   and inheritance, 764
   interface, 764
   memory leak, 177, 485
null pointer, 176
runs destructor, 485
delete [], 135
and dynamically allocated array, 137
deque, 353
   as element type, 311
   assign, 328
   assignment (=), 328
   at, 325
   back, 324
   begin, 317
clear, 327
const_iterator, 316
const_reference, 317
const_reverse_iterator, 316
constructor from element count, uses
   copy constructor, 478
constructors, 307–309
difference_type, 316
element type constraints, 309, 323
empty, 323
end, 317
erase, 326
   invalidates iterator, 326
front, 324
insert, 319
   invalidates iterator, 320
iterator, 316
iterator supports arithmetic, 312
performance characteristics, 334
pop_back, 326
pop_front, 326
push_back, 318
   invalidates iterator, 321
push_front, 318
   invalidates iterator, 321
random-access iterator, 417
rbegin, 317, 412
reference, 317
relational operators, 321
rend, 317, 412
resize, 323
reverse_iterator, 316, 412
size, 323
size_type, 316
subscript ([ ]), 325
supports relational operators, 313
swap, 329
types defined by, 316
value_type, 317
dereference (*), 98, 108, 146, 389
and increment, 163
auto_ptr, 704
iterator, 98
on map iterator yields pair, 362
overloaded operator, 524
pointer, 119
yields lvalue, 99, 120
derivation list, class, 563, 621
access control, 570
default access label, 574
derived class, 285, 302, 558, 621
see also virtual function
access control, 561, 572
as base class, 566
assigned or copied to base object, 578
assignment (=), 586
constructor, 581
calls virtual function, 589
for remote virtual base, 744
initializes immediate base class, 583
constructor initializer list, 582
conversion to base, 567
access control, 579
copy constructor, 586
default derivation label, 574
definition, 563
destructor, 587
calls virtual function, 589
friendship not inherited, 576
handle class, 599
member operator delete, 764
member hides member in base, 593
multiple base classes, 731
no conversion from base, 580
scope, 590
scope (::) to access base class member, 569
static members, 576
using declaration
inherited functions, 593

member access, 574
with remote virtual base, 742
derived object
contains base part, 565
multiple base classes, contains base
part for each, 732
design
CachedObj, 766
class member access control, 563
class type conversions, 544
consistent definitions of equality and
relational operators, 520
is-a relationship, 573
Message class, 486
namespace, 714
of handle classes, 599
of header files, 67
export, 646
inclusion compilation model, 644
separate compilation model, 645
optimizing new and delete, 764
using freelist, 766
overloaded operator, 510–513
overview of use counting, 495
Query classes, 609–611
Queue, 647
resource allocation is initialization,
700–701
Sales_item handle class, 599
TextQuery class, 380
vector memory allocation strategy,
756
writing generic code, 634
pointer template argument, 671
destruction, order of, 749
derived objects, 587
multiple base classes, 733
virtual base classes, 747
deructor, 476, 484, 502
called during exception handling, 691
container elements, 485
definition, 486
derived class, 587
explicit call to, 762
implicitly called, 484
library classes, 709
Message, 491
multiple inheritance, 737
order of destruction, 485
derived objects, 587
multiple base classes, 733
virtual base classes, 747
resource deallocation, 700
rule of three, 485
exception for virtual destructors, 588
should not throw exception, 692
synthesized, 485, 486
use counting, 495, 497
valuelike classes, 500
virtual, multiple inheritance, 736
virtual in base class, 587
development environment, integrated, 3
difference_type, 101, 107, 316
dimension, 110

direct base class, see immediate base class, 621
direct-initialization, 48
using constructor, 477
directive, using, 721, 751
pitfalls, 724
Disc_item, 583
class definition, 583
discriminant, 794, 806
divides<T>, 534
division (/), 150
do while statement, 210
condition in, 211
domain_error, 219
dot operator (.), 25, 30
class member access, 445
double, 37
literal (numE or numenue), 39
long double, 37
notation output format control, 830
output format control, 828
duplicate word program, 400–404
revisited, 531
dynamic binding, 559, 621
in C++, 569
requirements for, 566
dynamic type, 568, 622
dynamic_cast, 183, 773, 806
example, 773
throws bad_cast, 774
to pointer, 773
to reference, 774
dynamically allocated, 145
const object, 177
array, 134, 134–139
delete, 137
definition, 135
initialization, 136
of const, 136
C-style string, 138
memory and object construction, 754
object, 174
auto_ptr, 702
constructor, 453
destructor, 485
exception, 700
E
edit-compile-debug, 16, 28
errors at link time, 635
else, see if statement
   dangling, 198, 223
empty
   string, 83, 107
   vector, 93, 107
   associative container, 359
   container, 323
   priority_queue, 352
   queue, 352
   stack, 351
encapsulation, 432, 474
   advantages, 434
end, 353
   map, 369
   set, 372
   vector, 97
   container, 317
end-of-file, 19, 28, 835
   entering from keyboard, 20
Endangered, 731
end1, 8
   manipulator flushes the buffer, 291
ends, manipulator flushes the buffer, 291
enum keyword, 62
enumeration, 62, 74
   conversion to integer, 182
function matching, 274
enumerator, 62, 74
   conversion to integer, 182
evironment, integrated development, 3
eof, 289
eofbit, 289
equal, 814
equal member function, 778
equal_range, 814
   associative container, 379
   book finding program, 379
equal_to<T>, 534
equality (==), 30, 154
string, 85
algorithm, 421
and assignment, 161
container, 322
container adaptor, 350
iterator, 98, 312
overloaded operator, 512, 518
consistent with equality, 520
string, 347
erase
associative container, 359
container, 402
invalidates iterator, 326
map, 368
multimap, 376
multiset, 376
sequential container, 326
set, 372
string, 340
derror, standard, 6
escape sequence, 40, 74
hexadecimal (\Xnnn), 40
octal (\nnn), 40
evaluation
order of, 149, 189
short-circuit, 152
exception, raise, see throw
exception
class, 216, 224
class hierarchy, 698
constructor, 220
extending the hierarchy, 697
header, 219
what member, 218, 697
exception handling, 215–220, 749
see also throw
see also catch clause
and terminate, 219
compared to assert, 221
exception in destructor, 692
finding a catch clause, 693
function try block, 696, 749
handler, see catch clause
library class destructors, 709
local objects destroyed, 691
specifier, 217, 224, 693, 749
nonreference, 693
reference, 694
types related by inheritance, 694
stack unwinding, 691
uncaught exception, 692
unhandled exception, 219
exception object, 690, 749
array or function, 689
initializes catch parameter, 693
must be copyable, 689
pointer to local object, 690
rethrow, 695
exception safety, 700, 749
exception specification, 706, 749
throw(), 708
function pointers, 711
unexpected, 708
violation, 708
virtual functions, 710
executable file, 4
EXIT_FAILURE, 247
EXIT_SUCCESS, 247
explicit constructor, 462, 474
copy-initialization, 477
export, 645
and header design, 646
keyword, 645, 683
exporting C++ to C, 802
expression, 7, 28, 148, 189
and operand conversion, 179
compound, 168, 188
constant, 62, 74
throw, 689, 750
expression statement, 192, 224
extended_compute, 542
extension, compiler, 145
extern, 53
extern ‘C’, see linkage directive
extern const, 57

F
factorial program, 250
fail, 289
failbit, 289
file
executable, 4
object, 68
source, 4, 29
file mode, 296, 302
combinations, 298
example, 299
file static, 719, 749
fill, 816
fill_n, 815
find, 392, 812
    book finding program, 377
    map, 368
    multimap, 377
    multiset, 377
    set, 372
    string, 344
find last word program, 414
find_first_of, 396, 812
find_first_not_of, string, 346
find_end, 812
find_first_of, 396, 812
    string, 345
find_if, 421, 812
find_last_not_of, string, 346
find_last_of, string, 346
find_val program, 234
fixed manipulator, 830
float, 37
    literal (numF or numE), 39
floating point
    notation output format control, 830
    output format control, 828
floating point literal, see double literal
flow of control, 192, 224
flush, manipulator flushes the buffer, 291
Folder, see Message
for statement, 29, 207
    condition in, 207
    execution flow, 208
    expression, 207
    for header, 207
    initialization statement, 207
    scope, 15
for statement
    for statement, 14
for_each, 813
form state, 825
forward declaration of class type, 438
forward iterator, 417, 424
fp_compute, 542
free store, 135, 145
freelist, 766, 806
friend, 465, 474
    class, 465
    class template
        explicit template instantiation, 657
        nontemplate class or function, 656
        template class or function, 656
        function template, example, 659
    inheritance, 575
member function, 466
overloaded function, 467
overloaded operator, 509
scope considerations, 466
    namespaces, 727
    template example, 658
friend keyword, 465
front
    queue, 352
    sequential container, 324
front_inserter, 406, 424
    compared to inserter, 406
fstream, 285, 293–299, 302
    see also istream
    see also ostream
    close, 294
    constructor, 293
    file marker, 838
    file mode, 296
    combinations, 298
    example, 299
    file random access, 838
    header, 285, 293
    off_type, 839
    open, 293
    pos_type, 839
    random IO sample program, 840
    seek and tell members, 838–842
function, 2, 29, 225, 281
    equal member, 778
    inline, 257, 281
    candidate, 270, 280
    conversion to pointer, 277
    and template argument, 639
    function returning, 228
    inline and header, 257
    member, 25, 29, see member function, 474
    nonvirtual, calls resolved at compile
        time, 569
overloaded, 265, 281
    compared to redeclaration, 266
    friend declaration, 467
    scope, 268
    virtual, 593
pure virtual, 596, 622
    example, 609
    recursive, 249, 281
    viable, 270, 282
    virtual, 559, 566–570, 622
    assignment operator, 588
calls resolved at run time, 568
compared to run-time type identification, 777
default argument, 570
derived classes, 564
destructor, 587
destructor and multiple inheritance, 736
exception specifications, 710
in constructors, 589
in destructor, 589
introduction, 561
multiple inheritance, 735
no virtual constructor, 588
overloaded, 593
overloaded operator, 615
overriding run-time binding, 570
return type, 564
run-time type identification, 772
scope, 594
to copy unknown type, 602
type-sensitive equality, 778
function adaptor, 533, 535, 553
bind1st, 535
bind2nd, 535
binder, 535
negator, 535
not1, 535
not2, 535
function body, 3, 29, 226, 281
function call
ambiguous, 269, 272
execution flow, 226
overhead, 257
through pointer to function, 278
through pointer to member, 784
to overloaded operator, 509
to overloaded postfix operator, 529
using default argument, 253
function declaration, 251
and header file, 252
exception specification, 708
function definition, 3
function matching, 269, 281
and overloaded function templates, 679–682
examples, 680
argument conversion, 269
conversion operator, 545
conversion rank, 272
class type conversions, 545
enumeration parameter, 274
integral promotion, 273
multiple parameters, 272
namespaces, 727
of member functions, 436
overloaded operator, 547–551
function name, 3, 29
function object, 531, 553
algorithms, 531
example, 534
binary, 533
library defined, 533
unary, 533
function pointer, 276–279
and template argument deduction, 640
definition, 276
exception specifications, 711
function returning, 228
initialization, 277
overloaded functions, 279
parameter, 278
return type, 278
typedef, 276
function prototype, 251, 281
function return type, 226, 227, 281
const reference, 249
no implicit return type, 228
nonreference, 247
uses copy constructor, 478
reference, 248
reference yields lvalue, 249
void, 245
function scope, 227
function table, 785
pointer to member, 785
function template, 625, 683
see also template parameter
see also template argument
see also instantiation
as friend, 659
compiler error detection, 634
declaration, 629
error detection, 635
explicit template argument, 642
and function pointer, 643
specifying, 642
export, 645
inline, 626
instantiation, 637
template argument deduction, 637
type-dependent code, 634
function template specialization
  compared to overloaded function, 673
declaration, 672, 673
element, 672
namespaces, 730
scope, 674
function try block, 696, 749

G
g++, 4
gcd program, 226
generate, 816
generate_n, 815
generic algorithm, see algorithm
generic handle class, 667, 683
generic memory management, see CachedObj
generic programming, 95, 624
  and pointer template argument, 671
type-independent code, 634
generate, 82, 107
element, 300, 386
global namespace, 716, 750
global scope, 54, 74
global variable, lifetime, 254
GNU compiler, 4
good, 289
goto statement, 214, 224
greater-than (>, 30, 153
greater-than-or-equal (>=), 30, 153
greater<T>, 534
greater_equal<T>, 534
GT6 program, 403
GT_cls, 532
guard header, 71, 74

H
.h file, 21
Handle, 667
  int instantiation, 668
  operations, 668
  Sales_item instantiation, 669
handle class, 599, 622
  copy control, 601
  copying unknown type, 602
design, 599
generic, 667, 683
  that hides inheritance hierarchy, 610
  using a, 603
handler, see catch clause
has-a relationship, 573
HasPtr
  as a smart pointer, 495
  using synthesized copy control, 493
  with value semantics, 499
header, 7, 29, 67, 74
  bitset, 101
cctype, 88, 107
cstddef, 104
iomanip, 829
string, 80
vector, 90
algorithm, 395
  and constant expression, 69
  and library names, 810
  C library, 90
cassert, 221
class definition, 264, 437
cstddef, 123
cstdlib, 247
cstring, 132
default argument, 254
deque, 307
design, 67
  export, 646
  inclusion compilation model, 644
  namespace members, 714
  separate compilation model, 645
exception, 219
fstream, 285, 293
function declaration, 252
inline function, 257
inline member function definition, 437
iostream, 285
iterator, 399
list, 307
map, 360, 375
new, 219
numeric, 395
programmer-defined, 67–72
queue, 349
Sales_item, 21, 67, 264
set, 373, 375
sstream, 285, 300
stack, 349
stdexcept, 217, 219
type_info, 219
using declaration, 80
utility, 356
<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>vector</td>
<td>307</td>
</tr>
<tr>
<td>header file, naming convention</td>
<td>264</td>
</tr>
<tr>
<td>header guard, 71, 74</td>
<td></td>
</tr>
<tr>
<td>heap, 135, 145</td>
<td></td>
</tr>
<tr>
<td>hex manipulator, 827</td>
<td></td>
</tr>
<tr>
<td>hexadecimal, literal (0Xnum or 0xnum)</td>
<td>38</td>
</tr>
<tr>
<td>hexadecimal escape sequence (\Xnnn)</td>
<td>40</td>
</tr>
<tr>
<td>hides, names in base hidden by names in derived</td>
<td>592</td>
</tr>
<tr>
<td>hierarchy, inheritance, 558, 566, 622</td>
<td></td>
</tr>
<tr>
<td>high-order bits, 102, 107</td>
<td></td>
</tr>
<tr>
<td>IDE, 3</td>
<td></td>
</tr>
<tr>
<td>identification, run-time type, 772–780, 807</td>
<td></td>
</tr>
<tr>
<td>identifier, 46, 74</td>
<td></td>
</tr>
<tr>
<td>naming convention, 47</td>
<td></td>
</tr>
<tr>
<td>reserved, 47</td>
<td></td>
</tr>
<tr>
<td>if statement, else branch, 224</td>
<td></td>
</tr>
<tr>
<td>if statement, 17, 29, 195, 224</td>
<td></td>
</tr>
<tr>
<td>compared to switch, 199</td>
<td></td>
</tr>
<tr>
<td>dangling else, 198</td>
<td></td>
</tr>
<tr>
<td>else branch, 18, 197</td>
<td></td>
</tr>
<tr>
<td>ifstream, 285, 293–299</td>
<td></td>
</tr>
<tr>
<td>see also istream</td>
<td></td>
</tr>
<tr>
<td>close, 294</td>
<td></td>
</tr>
<tr>
<td>constructor, 293</td>
<td></td>
</tr>
<tr>
<td>file marker, 838</td>
<td></td>
</tr>
<tr>
<td>file mode, 296</td>
<td></td>
</tr>
<tr>
<td>combinations, 298</td>
<td></td>
</tr>
<tr>
<td>example, 299</td>
<td></td>
</tr>
<tr>
<td>file random access, 838</td>
<td></td>
</tr>
<tr>
<td>off_type, 839</td>
<td></td>
</tr>
<tr>
<td>open, 293</td>
<td></td>
</tr>
<tr>
<td>pos_type, 839</td>
<td></td>
</tr>
<tr>
<td>random IO sample program, 840</td>
<td></td>
</tr>
<tr>
<td>seek and tell members, 838–842</td>
<td></td>
</tr>
<tr>
<td>immediate base class, 566, 622</td>
<td></td>
</tr>
<tr>
<td>implementation, 63, 74, 432</td>
<td></td>
</tr>
<tr>
<td>implementation inheritance, 573</td>
<td></td>
</tr>
<tr>
<td>implicit this pointer, overloaded oper- tor, 508</td>
<td></td>
</tr>
<tr>
<td>implicit conversion, see conversion, 189</td>
<td></td>
</tr>
<tr>
<td>implicit return, 245</td>
<td></td>
</tr>
<tr>
<td>from main allowed, 247</td>
<td></td>
</tr>
<tr>
<td>implicit this pointer, 260, 281, 431, 440</td>
<td></td>
</tr>
<tr>
<td>in and overloaded operator, 483</td>
<td></td>
</tr>
<tr>
<td>static member functions, 469</td>
<td></td>
</tr>
<tr>
<td>in (file mode), 296</td>
<td></td>
</tr>
<tr>
<td>include, see #include</td>
<td></td>
</tr>
<tr>
<td>includes, 822</td>
<td></td>
</tr>
<tr>
<td>inclusion compilation model, 644, 683</td>
<td></td>
</tr>
<tr>
<td>incomplete type, 438, 474</td>
<td></td>
</tr>
<tr>
<td>restriction on use, 438, 566, 693</td>
<td></td>
</tr>
<tr>
<td>increment (++) , 13, 30, 146, 190</td>
<td></td>
</tr>
<tr>
<td>reverse iterator, 412</td>
<td></td>
</tr>
<tr>
<td>and dereference, 163</td>
<td></td>
</tr>
<tr>
<td>iterator, 98, 108, 312</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 526</td>
<td></td>
</tr>
<tr>
<td>pointer, 114</td>
<td></td>
</tr>
<tr>
<td>prefix yields lvalue, 162</td>
<td></td>
</tr>
<tr>
<td>indentation, 14, 197</td>
<td></td>
</tr>
<tr>
<td>index, 87, 107</td>
<td></td>
</tr>
<tr>
<td>indirect base class, 566, 622</td>
<td></td>
</tr>
<tr>
<td>inequality (!), 30, 154</td>
<td></td>
</tr>
<tr>
<td>iterator, 98</td>
<td></td>
</tr>
<tr>
<td>container, 322</td>
<td></td>
</tr>
<tr>
<td>container adaptor, 350</td>
<td></td>
</tr>
<tr>
<td>iterator, 312</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 512, 519</td>
<td></td>
</tr>
<tr>
<td>string, 347</td>
<td></td>
</tr>
<tr>
<td>inheritance, 284, 302</td>
<td></td>
</tr>
<tr>
<td>containers, 597</td>
<td></td>
</tr>
<tr>
<td>conversions, 577</td>
<td></td>
</tr>
<tr>
<td>default access label, 574</td>
<td></td>
</tr>
<tr>
<td>friends, 575</td>
<td></td>
</tr>
<tr>
<td>handle class, 599</td>
<td></td>
</tr>
<tr>
<td>implementation, 573</td>
<td></td>
</tr>
<tr>
<td>interface, 573</td>
<td></td>
</tr>
<tr>
<td>iostream diagram, 285</td>
<td></td>
</tr>
<tr>
<td>multiple, see multiple base class, 731</td>
<td></td>
</tr>
<tr>
<td>private, 622</td>
<td></td>
</tr>
<tr>
<td>static members, 576</td>
<td></td>
</tr>
<tr>
<td>virtual, 741, 751</td>
<td></td>
</tr>
<tr>
<td>inheritance hierarchy, 558, 566, 622</td>
<td></td>
</tr>
<tr>
<td>inheritance vs. composition, 573</td>
<td></td>
</tr>
<tr>
<td>initialization, 9</td>
<td></td>
</tr>
<tr>
<td>vs. assignment, 49</td>
<td></td>
</tr>
<tr>
<td>array, 111</td>
<td></td>
</tr>
<tr>
<td>array of char, 112</td>
<td></td>
</tr>
<tr>
<td>built-in type, 51</td>
<td></td>
</tr>
<tr>
<td>C-style string, 130</td>
<td></td>
</tr>
<tr>
<td>class data member, 454</td>
<td></td>
</tr>
<tr>
<td>class member of built-in type, 264</td>
<td></td>
</tr>
<tr>
<td>class member of class type, 264</td>
<td></td>
</tr>
<tr>
<td>class type, 52, 452</td>
<td></td>
</tr>
<tr>
<td>const static data member, 470</td>
<td></td>
</tr>
<tr>
<td>dynamically allocated array, 136</td>
<td></td>
</tr>
<tr>
<td>dynamically allocated object, 174</td>
<td></td>
</tr>
<tr>
<td>map, 373</td>
<td></td>
</tr>
</tbody>
</table>
memberwise, 479, 503
multi-dimensioned array, 142
objects of concrete class type, 464
pair, 356
parameter, 229
pointer, 117–119
pointer to function, 277
return value, 247
value, 92, 108
variable, 48, 50, 76
initialization vs. assignment, 456
initialized, 48, 75
initializer list, constructor, 263, 281, 431,
453–458, 474
inline function, 257, 281
and header, 257
function template, 626
member function, 437
and header, 437
inner_product, 823
inplace_merge, 816
input, standard, 6
input (>>, 8, 30
Sales_item, 516
istream_iterator, 408
string, 81, 108
overloaded operator, 515
error handling, 516–517
must be nonmember, 514
precedence and associativity, 158
input iterator, 416, 424
insert
inserter, 406
invalidates iterator, 320
map, 364
multimap, 376
multiset, 376
return type from set::insert, 373
sequential container, 319
set, 373
string, 340
insert iterator, 399, 405, 425
inserter, 406
inserter, 425
compared to front_inserter, 406
instantiation, 625, 684
class template, 628, 636, 654
member function, 653
nontype parameter, 655
type, 637
error detection, 635
function template, 637
from function pointer, 640
nontemplate argument conversion, 640
nontype template parameter, 633
template argument conversion, 638
member template, 663
nested class template, 788, 791
on use, 636
static class member, 665
int, 34
literal, 38
Integral, 539
integral promotion, 180, 189
function matching, 273
integral type, 34, 75
integrated development environment, 3
interface, 63, 75, 432
interface inheritance, 573
internal manipulator, 832
interval, left-inclusive, 314, 354
invalid_argument, 219
invalidated iterator, 315, 353
IO stream, see stream
iomanip header, 829
iostate, 289
iostream, 6, 29, 285
see also istream
see also ostream
header, 285
inheritance hierarchy, 740
seek and tell members, 838
is-a relationship, 573
isalnum, 88
isalpha, 88
ISBN, 2
isbn_mismatch, 699
destructor explained, 709
iscntrl, 88
isdigit, 88
isgraph, 88
islower, 88
isprint, 88
ispunct, 88
isShorter program, 235, 403
isspace, 88
iostream, 6, 29, 285
see also manipulator
condition state, 287
flushing input buffer, 290
format state, 825
Index

gcount, 837
get, 834
multi-byte version, 836
returns int, 835, 836
ggetline, 82, 836
ggetline, example, 300
ignore, 837
inheritance hierarchy, 740
input (\texttt{\textgreater\textless}), 8
precedence and associativity, 158
no containers of, 310
no copy or assign, 287
peek, 834
put, 834
putback, 835
read, 837
seek and tell members, 838
unformatted operation, 834
multi-byte, 837
single-byte, 834
unget, 835
write, 837
\texttt{istream\_iterator}, 407, 425
and class type, 410
constructors, 408
input iterator, 417
input operator (\texttt{\textgreater\textless}), 408
limitations, 411
operations, 409
used with algorithms, 411
\texttt{istringstream}, 285, 299–301
\textit{see also} \texttt{istream}
word per line processing, 300, 370
\texttt{str}, 301
word per line processing, 386
\texttt{isupper}, 88
\texttt{isxdigit}, 88
\texttt{Item\_base}
\hfill class definition, 560
\hfill constructor, 580
\hfill interface, 558
\hfill member functions, 559
\texttt{iter\_swap}, 816
\texttt{iterator}, 95, 95–101, 107, 311–316, 354
\hfill argument, 238
\hfill arrow (\texttt{\textgreater\textless}), 164
\hfill bidirectional, 417, 424
\hfill compared to reverse iterator, 413, 414
\hfill destination, 399, 419
\hfill equality, 98, 312
\hfill forward, 417, 424
\hfill generic algorithms, 394
\hfill inequality, 98, 312
\hfill input, 416, 424
\hfill insert, 399, 405, 425
\hfill invalidated, 315, 353
\hfill invalidated by
\hfill assign, 328
\hfill erase, 326
\hfill insert, 321
\hfill push\_back, 321
\hfill push\_front, 321
\hfill resize, 324
\hfill off-the-end, 97, 394, 425
\hfill operations, 311
\hfill output, 416, 425
\hfill parameter, 238, 242
\hfill random-access, 417, 425
\hfill relational operators, 313
\hfill reverse, 405, 412–415, 425
\hfill stream, 425
\hfill \texttt{iterator}, 362, 374
\hfill \texttt{container}, 316
\hfill \texttt{iterator\ arithmetic}, 399
\hfill \texttt{iterator\ arithmetic\ header}, 399
\hfill \texttt{iterator\ arithmetic\ hierarchy}, 399
\hfill \texttt{bidirectional\ iterator}, 399
\hfill \texttt{forward\ iterator}, 399
\hfill \texttt{input\ iterator}, 399
\hfill \texttt{output\ iterator}, 399
\hfill \texttt{random-access\ iterator}, 399
\hfill \texttt{iterator\ range}, 314, 314–316, 354
\hfill \texttt{algorithms\ constraints\ on}, 397, 415
\hfill \texttt{erase}, 327
\hfill \texttt{generic\ algorithms}, 394
\hfill \texttt{insert}, 320

\textbf{K}

\texttt{key\_type}, 388
\hfill \texttt{associative\ containers}, 362
\texttt{keyword}
\hfill \texttt{enum}, 62
\hfill \texttt{friend}, 465
\hfill \texttt{namespace}, 712
\hfill \texttt{protected}, 562
\hfill \texttt{template}, 625
\hfill \texttt{try}, 217
\hfill \texttt{union}, 793
virtual, 559
export, 645
keyword table, 46
Koenig lookup, 726

L
label
  case, 201, 201–204, 223
  access, 65, 73, 432, 473
  statement, 214
labeled statement, 214, 224
left manipulator, 832
left-inclusive interval, 314, 354
left-shift (<<), 155, 190
length_error, 219
less-than (<), 13, 30, 153
less<T>, 534
less_equal<T>, 534
lexicographical_compare, 823
library, standard, 5, 29
library names to header table, 810
library type, 29
lifetime, object, 254, 281
link time errors from template, 635
linkage directive, 801, 806
  C++ to C, 802
    compound, 802
    overloaded function, 803
    parameter or return type, 804
    pointer to function, 803
    single, 801
linking, 68, 75
list, 354
  as element type, 311
assign, 328
assignment (=), 328
back, 324
begin, 317
bidirectional iterator, 417
clear, 327
const_iterator, 316
const_reference, 317
const_reverse_iterator, 316
constructor from element count, uses
  copy constructor, 478
  constructors, 307–309
  element type constraints, 309, 323
empty, 323
end, 317
erase, 326
front, 324
insert, 319
iterator, 316
merge, 423
performance characteristics, 334
pop_back, 326
pop_front, 326
push_back, 318
push_front, 318
rbegin, 317, 412
reference, 317
relational operators, 321
remove, 423
remove_if, 423
rend, 317, 412
resize, 323
reverse, 423
reverse_iterator, 316, 412
size, 323
size_type, 316
specific algorithms, 421
splice, 423
swap, 329
types defined by, 316
unique, 423
value_type, 317

literal, 37, 37–42, 75
  bool, 39
  char, 40
  decimal, 38
  double (num\num or num\num\num), 39
  float (num\f or num\f\f), 39
  hexadecimal (0Xnum or 0xnum), 38
  int, 38
  long (num\L or num\L\L), 38
  long double (ddd.dddL or ddd.dddL\L), 39
  multi-line, 42
  octal (\num), 38
  string, 9, 30, 40
  unsigned (num\U or num\U\U), 39
  wchar_t, 40
local class, 796, 806
  access control, 796
  name lookup, 797
  nested class in, 797
  restrictions on, 796
local scope, 54, 75
<table>
<thead>
<tr>
<th>Index</th>
<th>865</th>
</tr>
</thead>
<tbody>
<tr>
<td>local static object, 255, 281</td>
<td></td>
</tr>
<tr>
<td>local variable, 227, 281</td>
<td></td>
</tr>
<tr>
<td>destructor, 485</td>
<td></td>
</tr>
<tr>
<td>lifetime, 254</td>
<td></td>
</tr>
<tr>
<td>reference return type, 248</td>
<td></td>
</tr>
<tr>
<td>logic_error, 219</td>
<td></td>
</tr>
<tr>
<td>logical AND (&amp;&amp;), 152</td>
<td></td>
</tr>
<tr>
<td>operand order of evaluation, 172</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 511</td>
<td></td>
</tr>
<tr>
<td>logical NOT (!), 153</td>
<td></td>
</tr>
<tr>
<td>logical operator, 152</td>
<td></td>
</tr>
<tr>
<td>function object, 533</td>
<td></td>
</tr>
<tr>
<td>logical OR (||), 152</td>
<td></td>
</tr>
<tr>
<td>operand order of evaluation, 172</td>
<td></td>
</tr>
<tr>
<td>overloaded operator, 511</td>
<td></td>
</tr>
<tr>
<td>logical_and&lt;T&gt;, 534</td>
<td></td>
</tr>
<tr>
<td>logical_not&lt;T&gt;, 534</td>
<td></td>
</tr>
<tr>
<td>logical_or&lt;T&gt;, 534</td>
<td></td>
</tr>
<tr>
<td>long, 34</td>
<td></td>
</tr>
<tr>
<td>literal (numL or numl), 38</td>
<td></td>
</tr>
<tr>
<td>long double, 37</td>
<td></td>
</tr>
<tr>
<td>long double, literal (ddd.dddL or ddd.dddL), 39</td>
<td></td>
</tr>
<tr>
<td>lookup, name, 447, 474</td>
<td></td>
</tr>
<tr>
<td>and templates, 647</td>
<td></td>
</tr>
<tr>
<td>before type checking, 269, 593</td>
<td></td>
</tr>
<tr>
<td>multiple inheritance, 738</td>
<td></td>
</tr>
<tr>
<td>class member declaration, 447</td>
<td></td>
</tr>
<tr>
<td>class member definition, 448, 450</td>
<td></td>
</tr>
<tr>
<td>class member definition, examples, 449</td>
<td></td>
</tr>
<tr>
<td>collisions under inheritance, 591</td>
<td></td>
</tr>
<tr>
<td>depends on static type, 590</td>
<td></td>
</tr>
<tr>
<td>multiple inheritance, 735</td>
<td></td>
</tr>
<tr>
<td>inheritance, 590, 595</td>
<td></td>
</tr>
<tr>
<td>local class, 797</td>
<td></td>
</tr>
<tr>
<td>multiple inheritance, 737</td>
<td></td>
</tr>
<tr>
<td>ambiguous names, 738</td>
<td></td>
</tr>
<tr>
<td>namespace names, 724</td>
<td></td>
</tr>
<tr>
<td>argument-dependent lookup, 726</td>
<td></td>
</tr>
<tr>
<td>nested class, 791</td>
<td></td>
</tr>
<tr>
<td>overloaded virtual functions, 593</td>
<td></td>
</tr>
<tr>
<td>virtual inheritance, 743</td>
<td></td>
</tr>
<tr>
<td>low-order bits, 102, 107</td>
<td></td>
</tr>
<tr>
<td>lower_bound, 814</td>
<td></td>
</tr>
<tr>
<td>associative container, 377</td>
<td></td>
</tr>
<tr>
<td>book finding program, 378</td>
<td></td>
</tr>
<tr>
<td>lvalue, 45, 75</td>
<td></td>
</tr>
<tr>
<td>assignment, 160</td>
<td></td>
</tr>
<tr>
<td>dereference, 99</td>
<td></td>
</tr>
<tr>
<td>function reference return type, 249</td>
<td></td>
</tr>
<tr>
<td>prefix decrement, 162</td>
<td></td>
</tr>
<tr>
<td>prefix increment, 162</td>
<td></td>
</tr>
<tr>
<td>subscript, 88</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
</tr>
<tr>
<td>machine-dependent</td>
<td></td>
</tr>
<tr>
<td>bitfield layout, 798</td>
<td></td>
</tr>
<tr>
<td>char representation, 36</td>
<td></td>
</tr>
<tr>
<td>division and modulus result, 151</td>
<td></td>
</tr>
<tr>
<td>end-of-file character, 20</td>
<td></td>
</tr>
<tr>
<td>iostate type, 288</td>
<td></td>
</tr>
<tr>
<td>linkage directve language, 802</td>
<td></td>
</tr>
<tr>
<td>nonzero return from main, 247</td>
<td></td>
</tr>
<tr>
<td>pre-compiled headers, 67</td>
<td></td>
</tr>
<tr>
<td>random file access, 837</td>
<td></td>
</tr>
<tr>
<td>reinterpret_cast, 185</td>
<td></td>
</tr>
<tr>
<td>representation of enum type, 274</td>
<td></td>
</tr>
<tr>
<td>return from exception what operation, 220</td>
<td></td>
</tr>
<tr>
<td>signed and out-of-range value, 37</td>
<td></td>
</tr>
<tr>
<td>signed types and bitwise operators, 155</td>
<td></td>
</tr>
<tr>
<td>size of arithmetic types, 34</td>
<td></td>
</tr>
<tr>
<td>template compilation optimization, 645</td>
<td></td>
</tr>
<tr>
<td>terminate function, 219</td>
<td></td>
</tr>
<tr>
<td>type_info members, 779</td>
<td></td>
</tr>
<tr>
<td>vector memory allocation size, 331</td>
<td></td>
</tr>
<tr>
<td>volatile implementation, 799</td>
<td></td>
</tr>
<tr>
<td>magic number, 56, 75</td>
<td></td>
</tr>
<tr>
<td>main, 2, 29</td>
<td></td>
</tr>
<tr>
<td>arguments to, 243</td>
<td></td>
</tr>
<tr>
<td>not recursive, 251</td>
<td></td>
</tr>
<tr>
<td>return type, 3</td>
<td></td>
</tr>
<tr>
<td>return value, 2–5, 247</td>
<td></td>
</tr>
<tr>
<td>returns 0 by default, 247</td>
<td></td>
</tr>
<tr>
<td>make_pair, 358</td>
<td></td>
</tr>
<tr>
<td>make_plural program, 248</td>
<td></td>
</tr>
<tr>
<td>manip, 542</td>
<td></td>
</tr>
<tr>
<td>manipulator, 8, 29, 825</td>
<td></td>
</tr>
<tr>
<td>boolalpha, 826</td>
<td></td>
</tr>
<tr>
<td>dec, 827</td>
<td></td>
</tr>
<tr>
<td>fixed, 830</td>
<td></td>
</tr>
<tr>
<td>hex, 827</td>
<td></td>
</tr>
<tr>
<td>internal, 832</td>
<td></td>
</tr>
<tr>
<td>left, 832</td>
<td></td>
</tr>
<tr>
<td>noboolalpha, 827</td>
<td></td>
</tr>
<tr>
<td>noshowbase, 828</td>
<td></td>
</tr>
<tr>
<td>noshownpint, 832</td>
<td></td>
</tr>
<tr>
<td>noskipws, 833</td>
<td></td>
</tr>
</tbody>
</table>
nouppercase, 828
oct, 827
right, 832
scientific, 830
setfill, 832
setprecision, 829
setw, 832
showbase, 827
showpoint, 831
skipws, 833
uppercase, 828
boolalpha, 826
change format state, 825
dec, 827
der1 flushes the buffer, 291
dends flushes the buffer, 291
fixed, 830
flush flushes the buffer, 291
hex, 827
internal, 832
left, 832
noboolalpha, 827
noshowbase, 828
noshowpoint, 832
noskipws, 833
nouppercase, 828
oct, 827
right, 832
scientific, 830
setfill, 832
setprecision, 829
setw, 832
showbase, 827
showpoint, 831
skipws, 833
unitbuf flushes the buffer, 291
uppercase, 828
map, 356, 388
as element type, 311
assignment (=), 328
begin, 369
bidirectional iterator, 417
clear, 359
constructors, 360
count, 367
definition, 360
dereference yields pair, 362
element type constraints, 309
empty, 359
end, 369
equal_range, 379
erase, 359, 368
find, 368
header, 360
insert, 364
iterator, 362
key type constraints, 360
key_type, 362
lower_bound, 377
mapped_type, 362, 388
operations, 358
overriding the default comparison, 604
rbegin, 412
rend, 412
return type from insert, 365
reverse_iterator, 412
size, 359
subscript operator, 363
supports relational operators, 359
swap, 329
upper_bound, 377
value_type, 361
mapped_type, map, multimap, 362
match, best, 269, 280
max, 822
max_element, 822
member, see also class member
mutable data, 474
pointer to, 780, 807
member function, 25, 29, 431, 474
const, 280
equal, 778
as friend, 466
base member hidden by derived, 593
class template, 653
defined outside class body, 651
instantiation, 653
const, 261, 262
defined outside class body, 261, 431
definition, 258–262
in class scope, 445
name lookup, 448
name lookup, examples, 449
function template, see member template
implicitly inline, 259
inline, 437
and header, 437
overloaded, 436
overloaded on const, 442
overloaded operator, 483, 508
Index

pointer to, definition, 782
returning *this, 442
static, 467
this pointer, 469
undefined, 482
member operator delete, 764, 806
and inheritance, 764
eexample, 769
CachedObj, 770
interface, 764
member operator delete [], 765
member operator new, 764, 806
eexample, 769
CachedObj, 769
interface, 764
member operator new [], 765
member template, 660, 684
declaration, 661
defined outside class body, 662
eexamples, 660
instantiation, 663
template parameters, 663
memberwise assignment, 483, 503
memberwise initialization, 479, 503
memory exhaustion, 175
memory leak, 177, 485
after exception, 700
memory management, generic, see CachedObj
merge, 816
list, 423
Message, 486–491
assignment operator, 490
class definition, 488
copy constructor, 489
design, 486
destructor, 491
put_Msg_in_Folder, 489
remove_Msg_from_Folder, 491
method, see member function, 29
Microsoft compiler, 4
min, 822
min_element, 822
minus<T>, 534
mismatch, 814
mode, file, 296, 302
modulus (%), 151
modulus<T>, 534
multi-dimensioned array, 141–144
and pointer, 143
conversion to pointer, 143
definition, 142
initialization, 142
parameter, 241
subscript operator, 142
multi-line literal, 42
multimap, 375, 388
assignment (=), 328
begin, 369
clear, 359
constructors, 360
count, 377
dereference yields pair, 362
element type constraints, 309
empty, 359
equal_range, 379
erase, 359, 376
find, 377
has no subscript operator, 376
insert, 376
iterator, 362, 376
key type constraints, 360
key_type, 362
lower_bound, 377
mapped_type, 362
operations, 358, 376
overriding the default comparison, 604
rbegin, 412
rend, 412
return type from insert, 365
reverse_iterator, 412
size, 359
supports relational operators, 359
swap, 329
upper_bound, 377
value_type, 361
multiple base class, 750
see also virtual base class
ambiguities, 738
ambiguous conversion, 734
avoiding potential name ambiguities, 738
conversions, 734
definition, 731
destructor usually virtual, 736
name lookup, 737
object composition, 732
order of construction, 732
scope, 737
virtual functions, 735
multiple inheritance, see multiple base class,
731
multimap, 375, 388
assignment (=), 328
begin, 369
clear, 359
constructors, 360
count, 377
dereference yields pair, 362
element type constraints, 309
empty, 359
equal_range, 379
erase, 359, 376
find, 377
has no subscript operator, 376
insert, 376
iterator, 362, 376
key type constraints, 360
key_type, 362
lower_bound, 377
mapped_type, 362
operations, 358, 376
overriding the default comparison, 604
rbegin, 412
rend, 412
return type from insert, 365
reverse_iterator, 412
size, 359
supports relational operators, 359
swap, 329
upper_bound, 377
value_type, 361
multiple base class, 750
see also virtual base class
ambiguities, 738
ambiguous conversion, 734
avoiding potential name ambiguities, 738
conversions, 734
definition, 731
destructor usually virtual, 736
name lookup, 737
object composition, 732
order of construction, 732
scope, 737
virtual functions, 735
multiple inheritance, see multiple base class,
multiplication (*), 150
multiplies<T>, 534
multiset, 375, 388
   assignment (=), 328
begin, 372
clear, 359
constructors, 372
count, 377
element type constraints, 309
end, 372
equal_range, 379
erase, 359, 376
find, 377
insert, 376
iterator, 376
key type constraints, 360
lower_bound, 377
Sales_item, 605
operations, 358, 376
overriding the default comparison, 604
rbegin, 412
rend, 412
return type from insert, 373
reverse_iterator, 412
supports relational operators, 359
swap, 329
upper_bound, 377
   example, 607
value_type, 372
mutable data member, 443, 474

N
\n (newline character), 40
name lookup, 447, 474
   and templates, 647
   before type checking, 269, 593
   multiple inheritance, 738
class member declaration, 447
class member definition, 448, 450
class member definition, examples, 449
collisions under inheritance, 591
decls on static type, 590
multiple inheritance, 735
inheritance, 590, 595
local class, 797
multiple inheritance, 737
   ambiguous names, 738
namespace names, 724
   argument-dependent lookup, 726
   nested class, 791
   overloaded virtual functions, 593
   virtual inheritance, 743
name resolution, see name lookup
namespace, 8, 29, 712, 750
   class friend declaration scope, 727
   cplusplus_primer, 714
definition, 712
design, 714
discontiguous definition, 714
function matching, 727
global, 716
member, 713
member definition, 716
   outside namespace, 716
   restrictions, 716
nested, 717
scope, 713–717
unnamed, 718
   local to file, 718
   replace file static, 719
namespace keyword, 712
namespace alias, 720, 750
namespace pollution, 712, 750
naming convention
   header file, 264
   source file, 264
NDEBUG, 220
negate<T>, 534
negator, 535, 553
nested class, 786, 806
   access control, 787
   class defined outside enclosing class, 789
   in class template, 788
   in local class, 797
   member defined outside class body, 788
   name lookup, 791
QueueItem example, 787
relationship to enclosing class, 787, 790
scope, 786
static members, 790
union, 794
nested namespace, 717
nested type, see nested class 786
new, 145, 174, 189, 806
   compared to operator new, 760
   execution flow, 760
Index

header, 219
member operator, 806
member operator, interface, 764
placement, 761, 807
compared to construct, 762
new [], 135
new failure, 175
next_permutation, 821
noboolalpha manipulator, 827
NoDefault, 459
nonconst reference, 60
parameter, 232
limitations, 235
nonportable, 42
nonprintable character, 40, 75
nonreference
parameter, 230
uses copy constructor, 478
return type, 247
uses copy constructor, 478
nontype template parameter, 625, 628, 632, 684
see also template parameter
class template, 655
must be constant expression, 633
nonvirtual function, calls resolved at compile time, 569
noshowbase manipulator, 828
noshowpoint manipulator, 832
noskipws manipulator, 833
not equal, see inequality
not1, 535
not2, 535
not_equal_to<T>, 534
NotQuery, 609
definition, 616
eval function, 620
nouppercase manipulator, 828
nth_element, 818
NULL, 118
null pointer, 118
delete of, 176
null statement, 192, 224
null-terminated array, see C-style string
number, magic, 56, 75
numeric header, 395
numeric literal
float (numF or numf), 39
long (numL or numl), 39
long double (ddd.dddL or ddd.dddL), 39
unsigned (numU or numu), 39

O
object, 46, 75
automatic, 255, 280
function, 553
is not polymorphic, 569
local static, 255, 281
temporary, 247
object cleanup, see destructor
object creation
constructor, 452
order of construction, 456
derived objects, 581, 582
multiple base classes, 732
virtual base classes, 745
order of destruction, 485
derived objects, 587
multiple base classes, 733
virtual base classes, 747
object file, 68
object lifetime, 254, 281
and destructor, 485
compared to scope, 254
object-oriented programming, 285, 302, 622
key ideas in, 558–560
oct manipulator, 827
teral (0num), 38
octal escape sequence (\nnn), 40
off-the-end iterator, 97, 394, 425
istream_iterator, 408
off-the-end pointer, 125
ofstream, 285, 293–299
see also ostream
close, 294
constructor, 293
file marker, 838
file mode, 296
combinations, 298
example, 299
file random access, 838
toff_type, 839
open, 293
pos_type, 839
random IO sample program, 840
seek and tell members, 838–842
open, 293
open_file, example of, 370, 383
open_file program, 299
operand, 148, 189
order of evaluation
  comma operator, 172
  conditional operator, 172
  logical operator, 172
operator, 148, 189
  sizeof, 167
typeid, 775, 807
addition (+), 150
  string, 86
  iterator, 101, 313
  pointer, 123
address-of (&), 115
arrow (->), 164
  class member access, 445
assignment ( = ), 13, 30, 159
  and conversion, 179
  and equality, 161
  container, 328
  multiple inheritance, 737
  pointer, 120
  string, 86
  to signed, 37
  to unsigned, 37
  yields lvalue, 160
binary, 148, 188
bitwise AND (&), 156
bitwise exclusive or (^), 156
bitwise not (~), 155
bitwise OR ( |), example, 290
call ( ), 30, 226
comma ( , ), 168
  operand order of evaluation, 172
coma ( , ), example, 289
compound assignment (e.g., += ), iterator, 313
compound assignment (e.g., = ), 13, 30, 161
  string, 86
  arithmetic, 162
bitwise, 162
conditional (? :), 165
  operand order of evaluation, 172
decrement (--)
  iterator, 312
  prefix yields lvalue, 162
  reverse iterator, 412
dereference (*), 98
  and increment, 163
  iterator, 98
  on map yields pair, 362
pointer, 119
  yields lvalue, 99, 120
division ( / ), 150
dot ( . ), 25, 30
  class member access, 445
equality ( == ), 30, 154
  string, 85
  algorithm, 421
  container, 322
  container adaptor, 350
  iterator, 98, 312
  string, 347
greater-than ( > ), 30, 153
greater-than-or-equal ( >= ), 30, 153
increment ( ++ ), 13, 30
  and dereference, 163
  iterator, 98, 312
  pointer, 114
  prefix yields lvalue, 162
  reverse iterator, 412
inequality ( != ), 30, 154
  container, 322
  container adaptor, 350
  iterator, 98, 312
  string, 347
input ( >> ), 8, 30
  Sales_item, 516
  istream_iterator, 408
  string, 81
  precedence and associativity, 158
left-shift ( << ), 155, 190
less-than ( < ), 30, 153
  used by algorithm, 420
  less-than-or-equal ( <= ), 13, 30, 153
logical AND ( && ), 152
  operand order of evaluation, 172
logical NOT ( ! ), 153
  logical OR ( || ), 152
  operand order of evaluation, 172
modulus ( % ), 151
multiplication ( * ), 150
output ( << ), 7, 30
  bitset, 106
  ostream_iterator, 408
  string, 81
  precedence and associativity, 158
overloaded, 189, 482, 503
pointer to member
  arrow ( ->* ), 783
  dot ( .* ), 783
right-shift ( >> ), 155, 190
Index

scope (::), 8, 30, 78
class member, 85, 445
container defined type, 317
member function definition, 262
to override name lookup, 449
shift, 155, 190
sizeof, 167
subscript ([ ])
bitset, 105
deque, 325
map, 363
string, 87
vector, 94, 325
and multi-dimensional array, 142
and pointer, 124
array, 113
valid subscript range, 88
yields lvalue, 88
subtraction (-), 150
iterator, 101, 313
pointer, 123
unary, 148, 189
unary minus (-), 150
operator alternative name, 46
operator delete function, 760, 806
compared to deallocate, 761
compared to delete expression, 760
operator delete member, 764
and inheritance, 764
element, 769
CachedObj, 770
interface, 764
operator delete () member, 765
operator new function, 760, 806
compared to allocate, 761
compared to new expression, 760
operator new member, 764
element, 769
CachedObj, 769
interface, 764
operator new () member, 765
operator overloading, see overloaded operator
options to main, 243
order of construction, 456, 749
derived objects, 581, 582
multiple base classes, 732
virtual base classes, 746
order of destruction, 485, 749
derived objects, 587
multiple base classes, 733
virtual base classes, 747
order of evaluation, 149, 189
comma operator, 172
conditional operator, 172
logical operator, 172
ordering, strict weak, 360, 389
OrQuery, 609
definition, 618
eval function, 619
ostream, 6, 29, 285
see also manipulator
condition state, 287
floatfield member, 831
flushing output buffer, 290
format state, 825
inheritance hierarchy, 740
no containers of, 310
no copy or assign, 287
not flushed if program crashes, 292
output (<<), 7
preference and associativity, 158
precision member, 829
seek and tell members, 838
tie member, 292
unsetf member, 831
ostream_iterator, 407, 425
and class type, 410
constructors, 408
limitations, 411
operations, 410
output iterator, 417
output operator (<<), 408
used with algorithms, 411
ostream_iterator, 407, 425
see also ostream
str, 301
out (file mode), 296
out_of_range, 219, 325
out_of__stock, 699
output, standard, 6
output (<<), 7, 30
bitset, 106
ostream_iterator, 407
string, 81, 108
overloaded operator, 513
formatting, 514
must be nonmember, 514
precedence and associativity, 158
Sales_item, 514
output iterator, 416, 425
overflow, 150
overflow_error, 219
overload resolution, see function matching
overloaded function, 265, 281
  using declarations, 728
  using directive, 729
  compared to redeclaration, 266
  compared to template specialization, 673
friend declaration, 467
linkage directive, 803
namespaces, 727
scope, 268
virtual, 593
overloaded member function, 436
  on const, 442
overloaded operator, 189, 482, 503
  << (output operator), 513
  * (dereference), 524
  & (address-of), 511
  -> (arrow operator), 525
  [ ] (subscript), 522
  reference return, 522
  () (call operator), 530
= (assignment), 476, 483, 520
  and copy constructor, 484
  check for self-assignment, 490
Message, 490
  reference return, 483, 521
  rule of three, 485
  use counting, 495, 498
  valuelike classes, 501
>> (input operator), 515
  error handling, 516–517
  must be nonmember, 514
  << (output operator)
  formatting, 514
  must be nonmember, 514
Sales_item, 514
&& (logical AND), 511
| | (logical OR), 511
, (comma operator), 511
addition (+), Sales_item, 517
ambiguous, 550
arithmetic operators, 517
  as virtual function, 615
  binary operator, 508
  candidate functions, 549
  compound assignment (e.g., +=), 511
  Sales_item, 521
consistency between relational and
  equality operators, 520
definition, 482, 506
definition, 510–513
equality operators, 512, 518
  explicit call to, 509
  explicit call to postfix operators, 529
  function matching, 547–551
  member and this pointer, 483
  member vs. nonmember function, 508, 512
  postfix increment (++) and decrement
    (--) operators, 528
  precedence and associativity, 507
  prefix increment (++) and decrement
    (--) operators, 527
  relational operators, 511, 520
  require class-type parameter, 507
  unary operator, 508
overloading, see overloaded function
operator, see overloaded operator

P

pair, 356, 388
  as return type from map::insert, 365
  as return type from set::insert, 373
default constructor, 357
definition, 356
initialize, 356
make_pair, 358
operations, 357
  public data members, 357
Panda, 731
  virtual inheritance, 741
parameter, 226, 227, 281
  array and buffer overflow, 242
  array type, 238–244
C-style string, 242
const, 231
column reference, 235
  overloading, 275
ellipses, 244
function pointer, 278
  linkage directive, 804
initialize, 229
iterator, 238, 242
library container, 237lifetime, 255
local copy, 230  
matching, 229  
ellipsis, 244  
template specialization, 673  
with class type conversion, 541  
multi-dimensional array, 241  
nonconst reference, 232  
nonreference type, 230  
uses copy constructor, 478  
of member function, 260  
vector type, 237  
passing, 229  
pointer to const, 231  
overloading, 275  
pointer to function, 278  
linkage directive, 804  
pointer to nonconst, 231  
pointer type, 231, 239  
reference  
to array type, 240  
to pointer, 236  
template, see template parameter  
and main, 243  
type checking  
and template argument, 638  
of reference to array, 240  
parameter list, 3, 29, 226, 228  
member function definition, 446  
parentheses, override precedence, 169  
partial specialization, 678, 684  
partial_sort, 818  
partial_sort_copy, 818  
partial_sum, 824  
pagination, 817  
placement new, 761, 807  
compared to construct, 762  
plus<T>, 534  
pointer, 114, 114–126, 146  
array, 122  
arrow (\texttt{->}), 164  
as initializer of vector, 140  
as parameter, 231  
assignment, 120  
char*, see C-style string  
class member copy control, 492–501  
copy constructor, 480  
destructor, 485  
strategies, 499  
compared to iterator, 114  
compared to reference, 121  
const, 128  
const pointer to const, 129  
container constructor from, 308  
conversion from derived to base, 567  
conversion from derived to multiple base, 734  
conversion to bool, 182  
conversion to void, 181  
dangling, 176, 188  
synthesized copy control, 494  
declaration style, 116–117  
definition, 115  
delete, 176  
dynamic_cast, example, 773  
function returning, 228  
imPLICIT this, 260, 281  
initialization, 117–119  
is polymorphic, 569  
multi-dimensional array, 143  
new, 174  
null, 118  
off-the-end, 125  
pitfalls with generic programs, 671  
reference parameter, 236  
relational operator, 132  
return type and local variable, 249  
smart, 495, 503, 553  
handle class, 599  
overloaded \texttt{->} (arrow operator) and \texttt{*} (dereference), 524  
overloaded (++) and (**), 526  
subscript operator, 124  
to pointer, 122  
typedef, 129  
typeid operator, 776  
uninitialized, 117  
volatile, 800  
pointer arithmetic, 123, 146  
pointer to const, 127  
argument, 231  
conversion from nonconst, 127  
parameter, 231  
overloading, 275  
pointer to function, 276–279  
definition, 276  
exception specifications, 711  
function returning, 228  
initialization, 277  
linkage directive, 803  
overloaded functions, 279  
parameter, 278  
return type, 278
<table>
<thead>
<tr>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>typedef, 276</td>
</tr>
<tr>
<td>pointer to member, 780, 807</td>
</tr>
<tr>
<td>and typedef, 783</td>
</tr>
<tr>
<td>arrow (\textasciitilde\textasciitilde\textasciitilde), 783</td>
</tr>
<tr>
<td>definition, 781</td>
</tr>
<tr>
<td>dot (\textasciitilde\textasciitilde\textasciitilde), 783</td>
</tr>
<tr>
<td>function pointer, 782</td>
</tr>
<tr>
<td>function table, 785</td>
</tr>
<tr>
<td>pointer to nonconst argument, 231</td>
</tr>
<tr>
<td>parameter, 231</td>
</tr>
<tr>
<td>polymorphism, 558, 622</td>
</tr>
<tr>
<td>compile time polymorphism via templates, 624</td>
</tr>
<tr>
<td>run time polymorphism in C++, 569</td>
</tr>
<tr>
<td>pop</td>
</tr>
<tr>
<td>priority_queue, 352</td>
</tr>
<tr>
<td>queue, 352</td>
</tr>
<tr>
<td>stack, 351</td>
</tr>
<tr>
<td>pop_back, sequential container, 326</td>
</tr>
<tr>
<td>pop_front, sequential container, 326</td>
</tr>
<tr>
<td>portable, 797</td>
</tr>
<tr>
<td>postfix decrement (\textasciitilde\textasciitilde\textasciitilde)</td>
</tr>
<tr>
<td>overloaded operator, 528</td>
</tr>
<tr>
<td>yields rvalue, 163</td>
</tr>
<tr>
<td>postfix increment (\textasciitilde\textasciitilde\textasciitilde)</td>
</tr>
<tr>
<td>and dereference, 163</td>
</tr>
<tr>
<td>overloaded operator, 528</td>
</tr>
<tr>
<td>precedence, 124, 146, 149, 168, 189</td>
</tr>
<tr>
<td>of assignment, 160</td>
</tr>
<tr>
<td>of conditional, 166</td>
</tr>
<tr>
<td>of dot and dereference, 164</td>
</tr>
<tr>
<td>of increment and dereference, 163</td>
</tr>
<tr>
<td>of IO operator, 158</td>
</tr>
<tr>
<td>of pointer to member and call operator, 782</td>
</tr>
<tr>
<td>overloaded operator, 507</td>
</tr>
<tr>
<td>pointer parameter declaration, 241</td>
</tr>
<tr>
<td>precedence table, 170</td>
</tr>
<tr>
<td>predicate, 402, 425</td>
</tr>
<tr>
<td>prefix decrement (\textasciitilde\textasciitilde\textasciitilde\textasciitilde\textasciitilde\textasciitilde), 163</td>
</tr>
<tr>
<td>overloaded operator, 527</td>
</tr>
<tr>
<td>yields lvalue, 162</td>
</tr>
<tr>
<td>prefix increment (\textasciitilde\textasciitilde\textasciitilde\textasciitilde\textasciitilde\textasciitilde)</td>
</tr>
<tr>
<td>and dereference, 163</td>
</tr>
<tr>
<td>overloaded operator, 527</td>
</tr>
<tr>
<td>yields lvalue, 162</td>
</tr>
<tr>
<td>preprocessor, 70, 75</td>
</tr>
<tr>
<td>directive, 7, 29</td>
</tr>
<tr>
<td>macro, 221, 224</td>
</tr>
<tr>
<td>variable, 71</td>
</tr>
<tr>
<td>prev_permutation, 821</td>
</tr>
<tr>
<td>preventing copies of class objects, 481</td>
</tr>
<tr>
<td>print_total, 559</td>
</tr>
<tr>
<td>explained, 568</td>
</tr>
<tr>
<td>printable character, 88</td>
</tr>
<tr>
<td>printValues program, 240, 242, 243</td>
</tr>
<tr>
<td>priority_queue, 348, 354</td>
</tr>
<tr>
<td>constructors, 349</td>
</tr>
<tr>
<td>relational operator, 350</td>
</tr>
<tr>
<td>private</td>
</tr>
<tr>
<td>class, 496</td>
</tr>
<tr>
<td>copy constructor, 481</td>
</tr>
<tr>
<td>inheritance, 571</td>
</tr>
<tr>
<td>member, 75, 474</td>
</tr>
<tr>
<td>private access label, 65, 432</td>
</tr>
<tr>
<td>inheritance, 561</td>
</tr>
<tr>
<td>private inheritance, 622</td>
</tr>
<tr>
<td>program</td>
</tr>
<tr>
<td>factorial, 250</td>
</tr>
<tr>
<td>find_val, 234</td>
</tr>
<tr>
<td>gcd, 226</td>
</tr>
<tr>
<td>is_Shorter, 235</td>
</tr>
<tr>
<td>make_plural, 248</td>
</tr>
<tr>
<td>open_file, 299</td>
</tr>
<tr>
<td>print_Values, 240, 242, 243</td>
</tr>
<tr>
<td>ptr_swap, 237</td>
</tr>
<tr>
<td>rgcd, 250</td>
</tr>
<tr>
<td>swap, 233, 245</td>
</tr>
<tr>
<td>vector capacity, 331</td>
</tr>
<tr>
<td>book finding</td>
</tr>
<tr>
<td>using equal_range, 379</td>
</tr>
<tr>
<td>using find, 377</td>
</tr>
<tr>
<td>using upper_bound, 378</td>
</tr>
<tr>
<td>bookstore, 26</td>
</tr>
<tr>
<td>bookstore exception classes, 698</td>
</tr>
<tr>
<td>Cached_Obj, 766</td>
</tr>
<tr>
<td>duplicate words, 400–404</td>
</tr>
<tr>
<td>revisited, 531</td>
</tr>
<tr>
<td>find last word, 414</td>
</tr>
<tr>
<td>GT6, 403</td>
</tr>
<tr>
<td>Handle class, 667</td>
</tr>
<tr>
<td>int instantiation, 668</td>
</tr>
<tr>
<td>operations, 668</td>
</tr>
<tr>
<td>Sales_item instantiation, 669</td>
</tr>
<tr>
<td>is_Shorter, 403</td>
</tr>
<tr>
<td>message handling classes, 486</td>
</tr>
<tr>
<td>Query</td>
</tr>
<tr>
<td>design, 609–611</td>
</tr>
<tr>
<td>interface, 610</td>
</tr>
<tr>
<td>operations, 607</td>
</tr>
<tr>
<td>Queue, 648</td>
</tr>
</tbody>
</table>
Index

- copy_elems member, 652
- destroy member, 651
- pop member, 651
- push member, 652
- random IO example, 840
- restricted word count, 374
- Sales_item handle class, 599
- Screen class template, 655
- TextQuery, 383
- class definition, 382
- design, 380
- interface, 381
- vector, capacity, 331
- vowel counting, 200
- word count, 363
- word transformation, 370
- ZooAnimal class hierarchy, 731
- programmer-defined header, 67–72
- programming
  - generic, 95, 624
  - object-oriented, 285, 302, 622
- promotion, integral, 180, 189
- protected, inheritance, 571, 622
- protected access label, 562, 622
- protected keyword, 562
- prototype, function, 251, 281
- ptr_swap program, 237
- ptrdiff_t, 123, 146
- public
  - inheritance, 571, 622
  - member, 75, 474
- public access label, 65, 432
- inheritance, 561
- pure virtual function, 596, 622
- example, 609
- push
  - priority_queue, 352
  - queue, 352
  - stack, 351
- push_back, 94, 108
  - vector, 94
  - back_inserter, 399
  - sequential container, 318
- push_front
  - front_inserter, 406
  - sequential container, 318
  - put_Msg_in_Folder, 489

Q

Query, 610

& (bitwise AND), 610
- definition, 614
- (bitwise NOT), 610
- definition, 614
| (bitwise OR), 610
- definition, 614
- (output operator), 615
- definition, 613
- design, 609–611
- interface, 610
- operations, 607
- Query_base, 609
- definition, 612
- member functions, 609
- Queue
  - (output operator), 659
  - assign, 662
  - copy_elems member, 652, 662
  - definition, 648
  - design, 647
  - destroy member, 651
  - final class definition, 664
  - interface, 627
  - member template declarations, 661
  - operations, 627
  - pop member, 651
  - push, specialized, 677
  - push member, 652
  - template version, char*, 675
- queue, 348, 354
  - constructors, 349
  - relational operator, 350
- QueueItem, 648
  - as nested class, 787
  - constructor, 789
  - definition, 788
  - friendship, 658
- CachedObj, 768
  - allocation explained, 769

R

Raccoon as virtual base, 741
RAII, see resource allocation is initialization
raise, 750
raise exception, see throw
random file IO, 838
random-access iterator, 417, 425
deque, 417
string, 417
vector, 417
random_shuffle, 820
range
  iterator, 314, 314–316, 354
  left-inclusive, 314
range_error, 219
rbegin, container, 317, 412
rdstate, 290
recursive function, 249, 281
refactoring, 583, 622
reference, 317
reference, 58, 75
  and pointer, 121
  const reference, 59
    initialization, 60
  conversion from derived to base, 567
  conversion from derived to multiple base, 734
dynamic_cast operator, example, 774
is polymorphic, 569
nonconst reference, 60
  parameter, 232–237
  pointer parameter, 236
  return type, is lvalue, 249
  return type and class object, 440
  return type and local variable, 249
  return value, 248
to array parameter, 240
reference count, see use count
reference data member, initialization, 455
reference to const, see const reference
reinterpret_cast, 183, 185
relational operator, 153
string, 85
  associative container, 359
  container, 321
  container adaptor, 350
  function object, 533
  overloaded operator, 511, 520
    consistent with equality, 520
  pointer, 132
remove, 819
  list, 423
remove_copy, 820
remove_copy_if, 820
remove_if, 819
  list, 423
remove_Msg_from_Folder, 491
rend, container, 317, 412
replace, 400, 816
string, 342
replace_copy, 400, 815
replace_copy_if, 815
replace_if, 816
reserve
  string, 336
  vector, 331
reserved identifier, 47
resize, sequential container, 323
Resource, 700
resource allocation is initialization, 700–701
auto_ptr, 702
restricted word count program, 374
result, 148, 189
rethrow, 695, 750
return, container, 381
return statement, 245–251
  from main, 247
  implicit, 245
  local variable, 247, 249
return type, 3, 29, 226, 227
  const reference, 249
  function, 281
  function pointer, 278
  linkage directive, 804
  member function definition, 446
  no implicit return type, 228
  nonreference, 247
    uses copy constructor, 478
    of virtual function, 564
  pointer to function, 278
  reference, 248
  reference yields lvalue, 249
  void, 245
return value
  conversion, 246
  copied, 247
reverse, 819
  list, 423
reverse iterator, 405, 412–415, 425
  -- (decrement), 412
  ++ (increment), 412
  base, 414
    compared to iterator, 413, 414
    example, 414
    requires -- (decrement), 413
reverse_copy, 819
reverse_iterator, 412
  container, 316
rfind, string, 346
## Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>rgcd program</td>
<td>250</td>
</tr>
<tr>
<td>right manipulator</td>
<td>832</td>
</tr>
<tr>
<td>right-shift (&gt;&gt;)</td>
<td>155, 190</td>
</tr>
<tr>
<td>rotate</td>
<td>819</td>
</tr>
<tr>
<td>rotate_copy</td>
<td>820</td>
</tr>
<tr>
<td>rule of three</td>
<td>485, 503</td>
</tr>
<tr>
<td>exception for virtual destructors</td>
<td>588</td>
</tr>
<tr>
<td>run time</td>
<td>75</td>
</tr>
<tr>
<td>error</td>
<td>38</td>
</tr>
<tr>
<td>run-time type identification</td>
<td>772–780, 807</td>
</tr>
<tr>
<td>classes with virtual functions</td>
<td>772</td>
</tr>
<tr>
<td>compared to virtual functions</td>
<td>777</td>
</tr>
<tr>
<td>dynamic_cast</td>
<td>773</td>
</tr>
<tr>
<td>example</td>
<td>773</td>
</tr>
<tr>
<td>throws bad_cast, 774</td>
<td></td>
</tr>
<tr>
<td>to pointer, 773</td>
<td></td>
</tr>
<tr>
<td>to reference, 774</td>
<td></td>
</tr>
<tr>
<td>type-sensitive equality</td>
<td>778</td>
</tr>
<tr>
<td>typeid</td>
<td>775</td>
</tr>
<tr>
<td>and virtual functions</td>
<td>775</td>
</tr>
<tr>
<td>example</td>
<td>776</td>
</tr>
<tr>
<td>returns type_info</td>
<td>776</td>
</tr>
<tr>
<td>runtime_error, 217, 219</td>
<td></td>
</tr>
<tr>
<td>constructor from string, 218</td>
<td></td>
</tr>
<tr>
<td>rvalue</td>
<td>45, 75</td>
</tr>
</tbody>
</table>

### S

safety, exception | 700 |

Sales_item | 21 |

addition (+), 23, 517 |
| throws exception | 217, 699 |
| class definition | 64, 258–265 |
| compare function | 604 |
| compound assignment (e.g., +=), 521 |
| conversion | 461 |
| default constructor | 263 |
| equality operators (==), (!=), 519 |
| explicit constructor | 462 |
| handle class | 599 |
| clone function | 602 |
| constructor | 601, 602 |
| definition | 600 |
| design | 599 |
| multiset of | 605 |
| using generic Handle, 669 |
| header | 21, 67, 264 |
| input (>>, 516 |
| istream constructor | 452 |
| no relational operators | 520 |
| operations | 21 |
| output (<<), 514 |
| avg_price definition | 261 |
| same_isbn, 24, 258 |
| string constructor | 452 |
| scientific manipulator | 830 |
| scope | 54, 75 |
| const object | 57, 69 |
| block | 193 |
| class, 65, 444, 473 |
| compared to object lifetime, 254 |
| for statement | 15 |
| friend declaration | 466 |
| function | 227 |
| function template specialization | 674 |
| global | 54, 74 |
| local | 54, 75 |
| multiple inheritance | 737 |
| namespace | 713–717 |
| statement | 194 |
| template parameter | 629 |
| using declaration | 720 |
| using directive | 721 |
| example | 722 |
| name collisions | 723 |

**scope (::)** |

base class members | 569 |
| namespace member | 750 |

**scope operator (::)** | 8, 30, 78, 108 |

class member | 85, 445 |
| container defined type | 317 |
| member function definition | 262 |
| namespace member | 713 |
| to override class-specific memory allocation | 765 |
| to override name lookup | 449 |

**Screen**, 435 |

class template | 655 |
| concatenating operations | 441 |
| display | 442 |
| do_display | 442 |
| friends | 465 |
| get definition | 446 |
| get members | 436 |
| get_cursor definition | 446 |
| CachedObj | 768 |
| Menu function table | 785 |
| move members | 441 |
| set members | 441 |
| simplified | 781 |
| size_type | 435 |
| ScreenPtr | 523 |
878  Index

arrow operator (\(\rightarrow\)), 525
derreference (*), 524
use counted, 523
ScrPtr, 523
search, 813
search_n, 813
self-assignment
    auto_ptr, 705
    check, 490
    use counting, 498
semantics, value, 499, 503
semicolon (;), 3
semicolon (;), class definition, 440
sentinel, 97, 108
separate compilation, 67, 76
    inclusion model for templates, 644
    of templates, 643
    separate compilation model for tem-
    plates, 645, 684
sequence, escape, 74
sequence (\(\backslash nnn\)), hexadecimal escape, 40
sequential container, 306, 354
    assign, 328
    assignment (=), 328
    back, 324
    clear, 327
    const_iterator, 316
    const_reverse_iterator, 316
    constructor from element count
        uses copy constructor, 478
        uses element default constructor, 460
    constructors, 307–309
deque, 306
    element type constraints, 309, 323
    empty, 323
    erase, 326
    front, 324
    insert, 319
    iterator, 316
    list, 306
    operations, 316–330
    performance characteristics, 333
    pop_back, 326
    pop_front, 326
    priority_queue, 348
    push_back, 318
    push_front, 318
    queue, 348
    rbegin, 412
    rend, 412
    resize, 323
    returning a, 381
    reverse_iterator, 316, 412
    size, 323
    size_type, 316
    stack, 348
    supports relational operators, 321
    swap, 329
    types defined by, 316
    value_type, 317
    vector, 306
set, 356, 388
    as element type, 311
    assignment (=), 328
    begin, 372
    bidirectional iterator, 417
    clear, 359
    constructors, 372
    count, 372
    element type constraints, 309
    empty, 359
    end, 372
    equal_range, 379
    erase, 359, 372
    find, 372
    insert, 373
    iterator, 374
    key type constraints, 360
    lower_bound, 377
    operations, 358
    overriding the default comparison,
       604
    rbegin, 412
    rend, 412
    return alternatives, 381
    return type from insert, 373
    reverse_iterator, 412
    size, 359
    supports relational operators, 359
    swap, 329
    upper_bound, 377
    value_type, 372
set_difference, 822
set_intersection, 619, 822
set_symmetric_difference, 822
set_union, 822
setfill manipulator, 832
setprecision manipulator, 829
setstate, 289, 290
setw manipulator, 832
shift operator, 155, 190
short, 34
short-circuit evaluation, 152
overloaded operator, 508
shorterString, 248
showbase manipulator, 827
showpoint manipulator, 831
signed, 35, 76
carcuation to unsigned, 36, 180
size, 108
string, 83
vector, 93
associative container, 359
priority_queue, 352
queue, 352
sequential container, 323
stack, 351
size_t, 104, 108, 146
and array, 113
size_type, 84, 108
string, 84
vector, 93
container, 316
sizeof operator, 167
skipws manipulator, 833
sliced, 579, 622
SmallInt, 536, 550
conversion operator, 537
smart pointer, 495, 503, 553
handle class, 599
overloaded -> (arrow operator) and
* (dereference), 524
overloaded (++) and (*), 526
sort, 401, 817
source file, 4, 29
naming convention, 264
specialization
class template
declaration, 677
definition, 675
member defined outside class body, 676
partial, 678
partial specialization, 684
class template member, 677
declaration, 677
function template
compared to overloaded function, 673
declaration, 672, 673
example, 672
scope, 674
template, namespaces, 730
specifier, type, 48, 76
splice, list, 423
sstream
header, 285, 300
str, 301
stable_partition, 817
stable_sort, 403, 817
stack, 348, 354
constructors, 349
relational operator, 350
stack unwinding, 691, 750
standard error, 6, 29
standard input, 6, 29
standard library, 5, 29
standard output, 6, 30
state, condition, 302
statement, 2, 30
break, 212, 223
continue, 214, 223
do while, 210
for, 29, 207
goto, 214, 224
if, 17, 29, 195, 224
return, 245–251
switch, 199, 224
while, 12, 30, 204, 224
compound, 193, 223
declaration, 193, 224
expression, 192, 224
for statement, 14
labeled, 214, 224
null, 192, 224
return, local variable, 247, 249
statement block, see block
statement label, 214
statement scope, 194
statement for statement, for, 14
static (file static), 719
static class member, 467, 474
as default argument, 471
class template, 665
accessed through an instantiation, 666
definition, 666
const data member, initialization, 470
const member function, 469
data member, 469
as constant expression, 471
inheritance, 576
member function, 467
Index

this pointer, 469
static object, local, 235, 281
static type, 568, 622
determines name lookup, 590
multiple inheritance, 735
static type checking, 44, 76
argument, 229
function return value, 246
static_cast, 183, 185
std, 8, 30
stdexcept header, 217, 219
store, free, 135, 145
str, 301
strcat, 133
strcmp, 133
strncpy, 133
stream
    istream_iterator, 407
    ostream_iterator, 407
    flushing buffer, 290
    iterator, 405, 407–412
    and class type, 410
    limitations, 411
    used with algorithms, 411
    not flushed if program crashes, 292
type as condition, 19
stream iterator, 425
strict weak ordering, 360, 389
string, C-style, see C-style string
string, 80–89
addition, 86
addition to string literal, 87
and string literal, 81, 140
append, 342
are case sensitive, 344
as sequential container, 335
assign, 340
assignment (=), 86
c_str, 140
c_str, example, 294
capacity, 336
compare, 347
compared to C-style string, 134
compound assignment, 86
concatenation, 86
constructor, 80, 338–339
default constructor, 52
def, 83
equality (==), 85
equality operator, 347
erase, 340
find, 344
find_first_not_of, 346
find_first_of, 345
find_last_not_of, 346
find_last_of, 346
getline, 82
ggetline, example, 300
header, 80
input operation as condition, 82
input operator, 81
insert, 340
output operator, 81
random-access iterator, 417
relational operator, 85, 347
replace, 342
reserve, 336
rfind, 346
size, 83
size_type, 84
subscript operator, 87
string literal, 9, 30, 40
addition to string, 87
and string library type, 81, 140
and C-style string, 140
concatenation, 41
stringstream, 285, 299–301, 302
    see also istream
    see also ostream
    str, 301
strlen, 133
strncat, 133
strncpy, 133
struct, see also class
default access label, 433
default inheritance access label, 574
struct, keyword, 66, 76, 474
in variable definition, 440
structure, data, 20, 28
Studio, Visual, 4
subscript ([ ]), 87, 108, 146, 389
bitset, 105
dequ, 325
map, 363
string, 87
vector, 94, 325
and multi-dimensioned array, 142
and pointer, 124
array, 113
overloaded operator, 522
reference return, 522
<table>
<thead>
<tr>
<th>Valid subscript range, 88</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yields lvalue, 88</td>
</tr>
<tr>
<td>Subscript range</td>
</tr>
<tr>
<td><em>string</em>, 88</td>
</tr>
<tr>
<td><em>vector</em>, 96</td>
</tr>
<tr>
<td><em>array</em>, 114</td>
</tr>
<tr>
<td><em>substr</em>, <em>string</em>, 342</td>
</tr>
<tr>
<td>Subtraction (−), 150</td>
</tr>
<tr>
<td><em>iterator</em>, 101, 313</td>
</tr>
<tr>
<td><em>pointer</em>, 123</td>
</tr>
<tr>
<td><em>swap</em>, 329, 816</td>
</tr>
<tr>
<td><em>container</em>, 329</td>
</tr>
<tr>
<td><em>swap_program</em>, 233, 245</td>
</tr>
<tr>
<td><em>swap_ranges</em>, 816</td>
</tr>
<tr>
<td>Switch statement, 199, 224</td>
</tr>
<tr>
<td>Default label, 203</td>
</tr>
<tr>
<td>And <em>break</em>, 201–203</td>
</tr>
<tr>
<td>Case label, 201</td>
</tr>
<tr>
<td>Compared to <em>if</em>, 199</td>
</tr>
<tr>
<td>Execution flow, 201</td>
</tr>
<tr>
<td>Expression, 203</td>
</tr>
<tr>
<td>Variable definition, 204</td>
</tr>
<tr>
<td>Synthesized assignment (=), 483, 503</td>
</tr>
<tr>
<td>Pointer members, 493</td>
</tr>
<tr>
<td>Synthesized copy constructor, 479, 503</td>
</tr>
<tr>
<td>Multiple inheritance, 737</td>
</tr>
<tr>
<td>Pointer members, 493</td>
</tr>
<tr>
<td>Virtual base class, 747</td>
</tr>
<tr>
<td>Synthesized copy control, volatile, 800</td>
</tr>
<tr>
<td>Synthesized default constructor, 264, 281, 459, 474</td>
</tr>
<tr>
<td>Inheritance, 581</td>
</tr>
<tr>
<td>Synthesized destructor, 485, 486</td>
</tr>
<tr>
<td>Multiple inheritance, 737</td>
</tr>
<tr>
<td>Pointer members, 493</td>
</tr>
</tbody>
</table>

**T**

\*t* (tab character), 40

Table of library name and header, 810

Template

*See also* class template

*See also* function template

*See also* instantiation

class, 90, 107

class member, see member template

link time errors, 635

overview, 624

Template keyword, 625

template argument, 625, 684

and function argument type checking, 638

class template, 628

conversion, 638

deduction, 684

*From function pointer, 640*

deduction for class template member function, 653

deduction for function template, 637

explicit and class template, 636

explicit and function template, 642

*And function pointer, 643*

specifying, 642

pointer, 671

Template argument deduction, 637

Template class, *see* class template

Template function, *see* function template

Template parameter, 625, 628–633, 684

*And member templates, 663*

name, 628

restrictions on use, 629

Nontype parameter, 625, 628, 632, 684

class template, 655

must be constant expression, 633

Scope, 629

type parameter, 625, 628, 630, 684

uses of inside class definition, 649

Template parameter list, 625, 684

Template specialization, 672, 684

Class member declaration, 677

*Compared to overloaded function, 673*

Definition, 675

Example, 672

Function declaration, 672, 673

Member defined outside class body, 676

Member of class template, 677

Parameter matching, 673

Partial specialization, 678, 684

Scope, 674

Template<> (template specialization)

temporary object, 247

Terminated, 219, 219, 224, 692, 750

TextQuery

class definition, 382

Main program using, 383

Program design, 380

Program interface, 381

Revisited, 609

This pointer

Implicit, 260, 281
implicit parameter, 431, 440
in overloaded operator, 483
overloaded operator, 508
static member functions, 469
three, rule of, 485, 503
throw, 216, 216, 224, 689, 750
example, 217, 290
execution flow, 218, 691
pointer to local object, 690
rethrow, 695
tolower, 88
top
priority_queue, 352
stack, 351
toupper, 88
transform, 815
transformation program, word, 370
translation unit, see source file
trunc (file mode), 296
try block, 216, 217, 224, 750
try keyword, 217
type
abstract data, 78, 473
arithmetic, 34, 73
built-in, 3, 28, 34–37
class, 20, 28, 65
compound, 58, 73, 145
dynamic, 568, 622
function return, 281
incomplete, 438, 474
integral, 34, 75
library, 29
nested, see nested class
return, 3, 29, 226, 227
static, 568, 622
determines name lookup, 590
name lookup and multiple inheritance, 735
type checking, 44
argument, 229
with class type conversion, 541
ellipsis parameter, 244
name lookup, 269
reference to array argument, 240
type identification, run-time, 772–780, 807
type specifier, 48, 76
type template parameter, 628, 630, 684
see also template parameter
type_info, 807
header, 219
name member, 780
no copy or assign, 780
operations, 779
returned from typeid, 776
typedef
and pointer, 129
and pointer to member, 783
pointer to function, 276
typedef, 61, 76
typeid operator, 775, 807
and virtual functions, 775
element, 776
returns type_info, 776
typename, keyword
compared to class, 631
in template parameter, 630
inside template definition, 632

U
U_Ptr, 496
unary function object, 533
unary minus (-), 150
unary operator, 148, 189
unary plus (+), 150
uncaught exception, 692
undefined behavior, 41, 76
dangling pointer, 176
synthesized copy control, 494
invalidated iterator, 315
uninitialized class data member, 459
uninitialized pointer, 117
uninitialized variable, 51
underflow_error, 219
unexpected, 708, 750
uninitialized, 8, 30, 51, 76
uninitialized pointer, 117
uninitialized_copy, 755, 759
uninitialized_fill, 755
union, 792, 807
anonymous, 795, 805
as nested type, 794
element, 794
limitations on, 793
union keyword, 793
unique, 402, 819
list, 423
unique_copy, 412, 820
unitbuf, manipulator flushes the buffer, 291
unnamed namespace, 718, 750
local to file, 718
Index 883

replace file static, 719
unsigned, 35, 76
  conversion to signed, 36, 180
  literal (numU or numu), 39
unsigned char, 36
unwinding, stack, 691, 750
upper_bound, 814
  associative container, 377
  book finding program, 378
  example, 607
uppercase manipulator, 828
use count, 495, 503
  design overview, 495
  generic class, 667
  held in companion class, 496
  pointer to, 600
  self-assignment check, 498
user, 433, 563
using declaration, 78, 108, 720, 750
  access control, 573
  class member access, 574
  in header, 80
  overloaded function, 728
  overloaded inherited functions, 593
  scope, 720
using directive, 721, 751
  overloaded function, 729
  pitfalls, 724
  scope, 721
  example, 722
  name collisions, 723
utility header, 356

V
value initialization, 92, 108
  map subscript operator, 363
vector, 92
  and dynamically allocated array, 136
dqueuendezqueune, 309
ulistlist, 309
  of dynamically allocated object, 175
  and resize, 324
  sequential container, 309
vvectorvector, 309
value semantics, 499, 503
value_type, 389
  map, multimap, 361
  sequential container, 317
set, multiset, 372
varargs, 244

variable, 8, 30, 43–55
  define before use, 44
  defined after case label, 204
  definition, 48
  definitions and goto, 215
  initialization, 48, 50, 76
    constructor, 452
  local, 227, 281
  scope, 55
Vector, 757
  capacity, 757
  memory allocation strategy, 757
  push_back, 758
  reallocate, 758
  size, 757
  using operator new and delete, 761
  using explicit destructor call, 763
  using placement new, 762
vector, 90–95, 354
  argument, 237
  as element type, 311
  assign, 328
  assignment (=), 328
  at, 325
  back, 324
  begin, 97, 317
  capacity, 331
  clear, 327
  const_iterator, 99, 316
  const_reference, 317
  const_reverse_iterator, 316
  constructor from element count, uses
    copy constructor, 478
  constructor taking iterators, 140
  constructors, 91–92, 307–309
  difference_type, 316
  element type constraints, 309, 323
  empty, 93, 323
  end, 97, 317
  erase, 326, 402
    invalidates iterator, 326
  front, 324
  header, 90
  initialization from pointer, 140
  insert, 319
    invalidates iterator, 320
  iterator, 97, 316
  iterator supports arithmetic, 312
  memory allocation strategy, 756
  memory management strategy, 330
index

parameter, 237
performance characteristics, 334
pop_back, 326
push_back, 94, 318
  invalidates iterator, 321
random-access iterator, 417
rbegin, 317, 412
reference, 317
relational operators, 321
rend, 317, 412
reserve, 331
resize, 323
 reverse_iterator, 316, 412
size, 93, 323
size_type, 93, 316
subscript (\[ \]), 325
subscript operator, 94
supports relational operators, 313
swap, 329
type, 91
types defined by, 316
value_type, 317
vector capacity program, 331
viable function, 270, 282
  with class type conversion, 545
virtual base class, 741, 751
  ambiguities, 743
  conversion, 743
  defining base as, 742
  derived class constructor, 744
  name lookup, 743
  order of construction, 746
  stream types, 741
virtual function, 559, 566–570, 622
  assignment operator, 588
  calls resolved at run time, 568
  compared to run-time type identification, 777
  default argument, 570
  derived classes, 564
destructor, 587
  multiple inheritance, 736
  exception specifications, 710
  in constructors, 589
  in destructor, 589
  introduction, 561
  multiple inheritance, 735
  no virtual constructor, 588
  overloaded, 593
  overloaded operator, 615
  overriding run-time binding, 570
pure, 596, 622
  example, 609
  return type, 564
  run-time type identification, 772
  scope, 594
  static, 469
  to copy unknown type, 602
type-sensitive equality, 778
virtual inheritance, 741, 751
virtual keyword, 559
Visual Studio, 4
void, 34, 76
  return type, 245
void*, 119, 146
  const void*, 127, 145
volatile, 800, 807
  pointer, 800
  synthesized copy control, 800
vowel counting program, 200

W
wchar_t, 34
  literal, 40
wchar_t streams, 286
wcin, 286
wcout, 286
weak ordering, strict, 360, 389
wfstream, 286
what, see exception
while statement, 12, 30, 204, 224
  condition in, 205
whitespace, 81
wide character streams, 286
wifstream, 286
window, console, 6
Window_Mgr, 465
wiostream, 286
wistream, 286
wistringstream, 300
wofstream, 286
wstringstream, 286
wofstream, 286
word, 35, 76
word count program, 363
  restricted, 374
word per line processing
  istream, 386
  istringstream, 370
  istream, 300
word transformation program, 370
WordQuery, 609
Index

definition, 616
wostream, 286
wostringstream, 286
wrap around, 38
wstringstream, 286

X
\Xmn (hexadecimal escape sequence), 40

Z
ZooAnimal, using virtual inheritance, 741
ZooAnimal class hierarchy, 731