C++:
The Complete Reference, Fourth Edition

Herbert Schildt

Tata McGraw-Hill Publishing Company Limited
NEW DELHI

McGraw-Hill Offices
New Delhi New York St Louis San Francisco Auckland Bogotá Caracas
Kuala Lumpur Lisbon London Madrid Mexico City Milan Montreal
San Juan Santiago Singapore Sydney Tokyo Toronto

S I T Library
Valachil, Mangalore

Accn No: M000994

Srinivas Institute of Technology
Acc. No. 994
Call No.

Tata McGraw-Hill

C++ : The Complete Reference, Fourth Edition

Copyright © 2003 by The McGraw-Hill Companies, Inc. All rights reserved. No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be produced foı publication

Tata McGraw-Hill Edition 2003

24th reprint 2009
RZZQCRLFRYZYA

Reprinted in India by arrangement with The McGraw-Hill Companies, Inc., New York

Sales territories: India, Pakistan, Nepal, Bangladesh, Sri Lanka and Bhutan

ISBN-13: 978-0-07-053246-5
ISBN-10: 0-07-053246-X

Published by Tata McGraw-Hill Publishing Company Limited,
7 West Patel Nagar, New Delhi 110 008, and printed at
Gopaljee Enterprises, Delhi 110 053

The McGraw·Hill Companies

Contents at a Glance

The Foundation of C++: The C Subset

1 An Overview of C 3
2 Expressions 13
3 Statements 57
4 Arrays and Null-Terminated Strings 89
5 Pointers 113
6 Functions 137
7 Structures, Unions, Enumerations, and User-Defined Types 161
8 C-Style Console I/O 187
9 File I/O 211
10 The Preprocessor and Comments 237

C++

11 An Overview of C++ 255
12 Classes and Objects 289

13 Arrays, Pointers, References, and the Dynamic Allocation Operators 325
14 Function Overloading, Copy Constructors, and Default Arguments 359
15 Operator Overloading 383
16 Inheritance 417
17 Virtual Functions and Polymorphism 443
18 Templates 459
19 Exception Handling 487
20 The C++ I/O System Basics 509
21 C++ File I/O 539
22 Run-Time Type ID and the Casting Operators 567
23 Namespaces, Conversion Functions, and Other Advanced Topics 591
24 Introducing the Standard Template Library 629

The Standard Function Library

25 The C-Based I/O Functions 699
26 The String and Character Functions 723
27 The Mathematical Functions 737
28 Time, Date, and Localization Functions 747
29 The Dynamic Allocation Functions 757
30 Utility Functions 761
31 The Wide-Character Functions 775

The Standard C++ Class Library

32 The Standard C++ I/O Classes 787
33 The STL Container Classes 811
34 The STL Algorithms 839
35 STL Iterators, Allocators, and Function Objects 861
36 The String Class 881
37 The Numeric Classes 897
38 Exception Handling and Miscellaneous Classes 925

Applying C++

39 Integrating New Classes: A Custom String Class 935
40 Parsing Expressions 963
A The .NET Managed Extensions to C++ 999
B C++ and the Robotics Age 1005
Index.. 1009

Contents

Introduction xxix

Part I

The Foundation of C++: The C Subset

1 An Overview of C 3

The Origins and History of C 4

C Is a Middle-Level Language 5

C Is a Structured Language 6

C Is a Programmer's Language 8

The Form of a C Program 9

The Library and Linking 10

Separate Compilation 12

Understanding the .C and .CPP File Extensions 12

2 Expressions 13

The Five Basic Data Types 14

Modifying the Basic Types 15

Identifier Names 16

Variables 17

Where Variables Are Declared 18

Local Variables 18

Formal Parameters 21
Global Variables 21
The const and volatile Qualifiers 23
const 23
volatile 24
Storage Class Specifiers 25
extern 25
static Variables 28
register Variables 30
Variable Initializations 31
Constants 32
Hexadecimal and Octal Constants 33
String Constants 33
Backslash Character Constants 33
Operators 35
The Assignment Operator 35
Type Conversion in Assignments 35
Multiple Assignments 37
Arithmetic Operators 37
Increment and Decrement 38
Relational and Logical Operators 40
Bitwise Operators 42
The ? Operator 47
The & and * Pointer Operators 48
The Compile-Time Operator sizeof 50
The Comma Operator 50
The Dot (.) and Arrow (–>) Operators 51
The [ ] and ( ) Operators 51
Precedence Summary 52
Expressions 52
Order of Evaluation 52
Type Conversion in Expressions 53
Casts 54
Spacing and Parentheses 55
Compound Assignments 56

3 Statements 57
True and False in C and C++ 58
Selection Statements 59
if 59
Nested ifs 60
The if-else-if Ladder 62
The ? Alternative 63
The Conditional Expression 66

switch . . . 67
Nested switch Statements . . . 70
Iteration Statements . . . 70
The for Loop . . . 70
for Loop Variations . . . 72
The Infinite Loop . . . 76
for Loops with No Bodies . . . 77
The while Loop . . . 77
The do-while Loop . . . 79
Declaring Variables Within Selection and Iteration Statements . . . 81
Jump Statements . . . 82
The return Statement . . . 82
The goto Statement . . . 83
The break Statement . . . 83
The exit( ) Function . . . 85
The continue Statement . . . 86
Expression Statements . . . 88
Block Statements . . . 88

4 Arrays and Null-Terminated Strings . . . 89
Single-Dimension Arrays . . . 90
Generating a Pointer to an Array . . . 92
Passing Single-Dimension Arrays to Functions . . . 92
Null-Terminated Strings . . . 94
Two-Dimensional Arrays . . . 96
Arrays of Strings . . . 100
Multidimensional Arrays . . . 101
Indexing Pointers . . . 102
Array Initialization . . . 105
Unsized Array Initializations . . . 106
A Tic-Tac-Toe Example . . . 108

5 Pointers . . . 113
What Are Pointers? . . . 114
Pointer Variables . . . 115
The Pointer Operators . . . 115
Pointer Expressions . . . 116
Pointer Assignments . . . 117
Pointer Arithmetic . . . 117
Pointer Comparisons . . . 119
Pointers and Arrays . . . 121
Arrays of Pointers . . . 122
Multiple Indirection . . . 123
Initializing Pointers . . . 125
Pointers to Functions . . . 126

C's Dynamic Allocation Functions 129
Problems with Pointers 131

6 Functions 137
The General Form of a Function 138
Scope Rules of Functions 138
Function Arguments 139
Call by Value, Call by Reference 139
Creating a Call by Reference 140
Calling Functions with Arrays 142
argc and argv—Arguments to main( ) 144
The return Statement 147
Returning from a Function 148
Returning Values 149
Returning Pointers 151
Functions of Type void 152
What Does main( ) Return? 153
Recursion 153
Function Prototypes 155
Standard Library Function Prototypes 157
Declaring Variable-Length Parameter Lists 158
Old-Style Versus Modern FunctionParameter Declarations 158

7 Structures, Unions, Enumerations,
and User-Defined Types 161
Structures 162
Accessing Structure Members 165
Structure Assignments 165
Arrays of Structures 166
Passing Structures to Functions 167
Passing Structure Members to Functions 167
Passing Entire Structures to Functions 168
Structure Pointers 169
Declaring a Structure Pointer 170
Using Structure Pointers 170
Arrays and Structures Within Structures 173
Bit-Fields 174
Unions 176
Enumerations 180
Using sizeof to Ensure Portability 183
typedef 184

8 C-Style Console I/O 187
An Important Application Note 188

Reading and Writing Characters 189
A Problem with getchar( ) 190
Alternatives to getchar( ) 190
Reading and Writing Strings 192
Formatted Console I/O 195
printf( ) 195
Printing Characters 196
Printing Numbers 196
Displaying an Address 198
The %n Specifier 198
Format Modifiers 199
The Minimum Field Width Specifier 199
The Precision Specifier 200
Justifying Output 201
Handling Other Data Types 202
The * and # Modifiers 202
scanf( ) 203
Format Specifiers 203
Inputting Numbers 203
Inputting Unsigned Integers 205
Reading Individual Characters Using scanf( ) 205
Reading Strings 205
Inputting an Address 206
The %n Specifier 206
Using a Scanset 206
Discarding Unwanted White Space 207
Non-White-Space Characters in the Control String 208
You Must Pass scanf( ) Addresses 208
Format Modifiers 208
Suppressing Input 209

9 File I/O 211
C Versus C++ File I/O 212
Streams and Files 212
Streams 212
Text Streams 213
Binary Streams 213
Files 213
File System Basics 214
The File Pointer 215
Opening a File 215
Closing a File 217
Writing a Character 217
Reading a Character 218

Using fopen( ), getc( ), putc( ), and fclose( ) 218
Using feof( ) 220
Working with Strings: fputs( ) and fgets( ) 222
rewind( ) 223
ferror( ) 224
Erasing Files 226
Flushing a Stream 226
fread( ) and fwrite( ) 227
Using fread( ) and fwrite( ) 227
fseek( ) and Random-Access I/O 229
fprintf( ) and fscanf( ) 230
The Standard Streams 232
The Console I/O Connection 233
Using freopen( ) to Redirect the Standard Streams 234

10 The Preprocessor and Comments 237
The Preprocessor 238
#define 238
Defining Function-like Macros 240
#error 241
#include 242
Conditional Compilation Directives 242
#if, #else, #elif, and #endif 243
#ifdef and #ifndef 245
#undef 246
Using defined 247
#line 248
#pragma 248
The # and ## Preprocessor Operators 248
Predefined Macro Names 250
Comments 250
Single-Line Comments 252

Part II
C++

11 An Overview of C++ 255
The Origins of C++ 256
What Is Object-Oriented Programming? 257
Encapsulation 258
Polymorphism 258
Inheritance 259
Some C++ Fundamentals 259
A Sample C++ Program 260

A Closer Look at the I/O Operators 263
Declaring Local Variables 264
No Default to int 265
The bool Data Type 266
Old-Style vs. Modern C++ 267
The New C++ Headers 268
Namespaces 269
Working with an Old Compiler 270
Introducing C++ Classes 270
Function Overloading 275
Operator Overloading 278
Inheritance 278
Constructors and Destructors 283
The C++ Keywords 287
The General Form of a C++ Program 288

12 Classes and Objects 289
Classes 290
Structures and Classes Are Related 293
Unions and Classes Are Related 295
Anonymous Unions 296
Friend Functions 297
Friend Classes 302
Inline Functions 303
Defining Inline Functions Within a Class 306
Parameterized Constructors 307
Constructors with One Parameter: A Special Case 309
Static Class Members 310
Static Data Members 310
Static Member Functions 315
When Constructors and Destructors Are Executed 317
The Scope Resolution Operator 319
Nested Classes 319
Local Classes 320
Passing Objects to Functions 320
Returning Objects 323
Object Assignment 324

13 Arrays, Pointers, References, and the Dynamic Allocation Operators 325
Arrays of Objects 326
Creating Initialized vs. Uninitialized Arrays 328
Pointers to Objects 329
Type Checking C++ Pointers 332

The this Pointer 332
Pointers to Derived Types 334
Pointers to Class Members 337
References 339
Reference Parameters 339
Passing References to Objects 343
Returning References 344
Independent References 345
References to Derived Types 346
Restrictions to References 347
A Matter of Style 347
C++'s Dynamic Allocation Operators 347
Initializing Allocated Memory 349
Allocating Arrays 350
Allocating Objects 351
The nothrow Alternative 356
The Placement Form of new 357

14 Function Overloading, Copy Constructors, and Default Arguments 359
Function Overloading 360
Overloading Constructors 362
Overloading a Constructor to Gain Flexibility 362
Allowing Both Initialized and Uninitialized Objects 364
Copy Constructors 366
Finding the Address of an Overloaded Function 370
The overload Anachronism 371
Default Function Arguments 371
Default Arguments vs. Overloading 376
Using Default Arguments Correctly 377
Function Overloading and Ambiguity 378

15 Operator Overloading 383
Creating a Member Operator Function 384
Creating Prefix and Postfix Forms of the Increment and Decrement Operators 389
Overloading the Shorthand Operators 390
Operator Overloading Restrictions 390
Operator Overloading Using a Friend Function 391
Using a Friend to Overload ++ or -- 393
Friend Operator Functions Add Flexibility 396
Overloading new and delete 398
Overloading new and delete for Arrays 403
Overloading the nothrow Version of new and delete 406

Overloading Some Special Operators 407
Overloading [ ] 407
Overloading ( ) 411
Overloading -> 413
Overloading the Comma Operator 414

16 Inheritance 417
Base-Class Access Control 418
Inheritance and protected Members 420
Protected Base-Class Inheritance 424
Inheriting Multiple Base Classes 425
Constructors, Destructors, and Inheritance 426
When Constructors and Destructors Are Executed 426
Passing Parameters to Base-Class Constructors 430
Granting Access 434
Virtual Base Classes 437

17 Virtual Functions and Polymorphism 443
Virtual Functions 444
Calling a Virtual Function Through a Base Class Reference 447
The Virtual Attribute Is Inherited 448
Virtual Functions Are Hierarchical 450
Pure Virtual Functions 453
Abstract Classes 455
Using Virtual Functions 455
Early vs. Late Binding 458

18 Templates 459
Generic Functions 460
A Function with Two Generic Types 463
Explicitly Overloading a Generic Function 463
Overloading a Function Template 466
Using Standard Parameters with Template Functions 466
Generic Function Restrictions 467
Applying Generic Functions 468
A Generic Sort 469
Compacting an Array 470
Generic Classes 472
An Example with Two Generic Data Types 476
Applying Template Classes: A Generic Array Class 477
Using Non-Type Arguments with Generic Classes 479
Using Default Arguments with Template Classes 481
Explicit Class Specializations 483

The typename and export Keywords 484
The Power of Templates 485

19 Exception Handling 487
Exception Handling Fundamentals 488
Catching Class Types 494
Using Multiple catch Statements 495
Handling Derived-Class Exceptions 497
Exception Handling Options 498
Catching All Exceptions 498
Restricting Exceptions 500
Rethrowing an Exception 502
Understanding terminate( ) and unexpected( ) 503
Setting the Terminate and Unexpected Handlers 504
The uncaught_exception( ) Function 505
The exception and bad_exception Classes 506
Applying Exception Handling 506

20 The C++ I/O System Basics 509
Old vs. Modern C++ I/O 510
C++ Streams 511
The C++ Stream Classes 511
C++'s Predefined Streams 512
Formatted I/O 513
Formatting Using the ios Members 513
Setting the Format Flags 514
Clearing Format Flags 515
An Overloaded Form of setf( ) 516
Examining the Formatting Flags 518
Setting All Flags 519
Using width( ), precision( ), and fill( ) 520
Using Manipulators to Format I/O 522
Overloading << and >> 526
Creating Your Own Inserters 526
Creating Your Own Extractors 532
Creating Your Own Manipulator Functions 535

21 C++ File I/O 539
<fstream> and the File Classes 540
Opening and Closing a File 540
Reading and Writing Text Files 543
Unformatted and Binary I/O 545
Characters vs. Bytes 545
put( ) and get( ) 546
read( ) and write( ) 548

More get( ) Functions 551
getline( ) 551
Detecting EOF 553
The ignore( ) Function 555
peek( ) and putback( ) 556
flush( ) 556
Random Access 557
Obtaining the Current File Position 561
I/O Status 561
Customized I/O and Files 563

22 Run-Time Type ID and the Casting Operators 567
Run-Time Type Identification (RTTI) 568
The Casting Operators 578
dynamic_cast 578

23 Namespaces, Conversion Functions, and Other Advanced Topics 591
Namespaces 592
Namespace Fundamentals 592
using 596
Unnamed Namespaces 598
Some Namespace Options 599
The std Namespace 601
Creating Conversion Functions 603
const Member Functions and mutable 607
Volatile Member Functions 609
Explicit Constructors 610
The Member Initialization Syntax 611
Using the asm Keyword 616
Linkage Specification 617
Array-Based I/O 618
The Array-Based Classes 619
Creating an Array-Based Output Stream 619
Using an Array as Input 621
Input/Output Array-Based Streams 623
Using Dynamic Arrays 624
Using Binary I/O with Array-Based Streams 625
Summarizing the Differences Between C and C++ 626

24 Introducing the Standard Template Library 629
An Overview of the STL 630
Containers 630

Algorithms 631
Iterators 631
Other STL Elements 632
The Container Classes 633
General Theory of Operation 634
Vectors 635
Accessing a Vector Through an Iterator 639
Inserting and Deleting Elements in a Vector 641
Storing Class Objects in a Vector 643
Lists 645
Understanding end( ) 649
push_front( ) vs. push_back( ) 651
Sort a List 652
Merging One List with Another 653
Storing Class Objects in a List 655
Maps 658
Storing Class Objects in a Map 662
Algorithms 664
Counting 664
Removing and Replacing Elements 670
Reversing a Sequence 672
Transforming a Sequence 673
Using Function Objects 675
Unary and Binary Function Objects 675
Using the Built-in Function Objects 675
Creating a Function Object 678
Using Binders 680
The string Class 683
Some string Member Functions 687
Strings Are Containers 693
Putting Strings into Other Containers 694
Final Thoughts on the STL 695

Part III

The Standard Function Library

25 The C-Based I/O Functions 699
clearerr 700
fclose 701
feof 701
ferror 701
fflush 702
fgetc 702
fgetpos 702

fgets 703
fopen 703
fprintf 705
fputc 705
fputs 706
fread 706
freopen 706
fscanf 707
fseek 707
fsetpos 708
ftell 708
fwrite 709
getc 709
getchar 710
gets 710
perror 710
printf 711
putc 714
putchar 714
puts 714
remove 715
rename 715
rewind 715
scanf 715
setbuf 719
setvbuf 719
sprintf 720
sscanf 720
tmpfile 720
tmpnam 721
ungetc 721
vprintf, vfprintf, and vsprintf 722

26 The String and Character Functions 723
isalnum 724
isalpha 724
iscntrl 725
isdigit 725
isgraph 725
islower 725
isprint 726
ispunct 726
isspace 726
isupper 727

isxdigit 727
memchr 727
memcmp 727
memcpy 728
memmove 728
memset 729
strcat 729
strchr 729
strcmp 730
strcoll 730
strcpy 731
strcspn 731
strerror 731
strlen 731
strncat 732
strncmp 732
strncpy 733
strpbrk 733
strrchr 733
strspn 734
strstr 734
strtok 734
strxfrm 735
tolower 735
toupper 735

27 The Mathematical Functions 737
acos 738
asin 738
atan 739
atan2 739
ceil 739
cos 740
cosh 740
exp 740
fabs 741
floor 741
fmod 741
frexp 741
ldexp 742
log 742
log10 742
modf 743
pow 743
sin 743

sinh 744
sqrt 744
tan 744
tanh 745

28 Time, Date, and Localization Functions 747
asctime 748
clock 749
ctime 749
difftime 750
gmtime 750
localeconv 750
localtime 752
mktime 752
setlocale 752
strftime 753
time 754

29 The Dynamic Allocation Functions 757
calloc 758
free 758
malloc 759
realloc 759

30 Utility Functions 761
abort 762
abs 762
assert 763
atexit 763
atof 763
atoi 764
atol 764
bsearch 764
div 765
exit 766
getenv 766
labs 766
ldiv 767
longjmp 767
mblen 767
mbstowcs 768
mbtowc 768
qsort 768
raise 769

rand 770
setjmp 770
signal 770
srand 771
strtod 771
strtol 772
strtoul 772
system 773
va_arg, va_start, and va_end 773
wcstombs 774
wctomb 774

31 The Wide-Character Functions 775
The Wide-Character Classification Functions 776
The Wide-Character I/O Functions 779
The Wide-Character String Functions 779
Wide-Character String Conversion Functions 779
Wide-Character Array Functions 782
Multibyte/Wide-Character Conversion Functions 783

Part IV
The Standard C++ Class Library

32 The Standard C++ I/O Classes 787
The I/O Classes 788
The I/O Headers 790
The Format Flags and I/O Manipulators 791
Several Data Types 793
The streamsize and streamoff Types 793
The streampos and wstreampos Types 793
The pos_type and off_type Types 793
The openmode Type 793
The iostate Type 794
The seekdir Type 794
The failure Class 794
Overload << and >> Operators 794
The General-Purpose I/O Functions 795
bad 795
clear 795
eof 795
exceptions 796
fail 796
fill 796
flags 797

flush 797
fstream, ifstream, and ofstream 797
gcount 798
get 798
getline 799
good 800
ignore 800
open 800
peek 801
precision 802
put 802
putback 802
rdstate 802
read 803
readsome 803
seekg and seekp 804
setf 805
setstate 805
str 806
stringstream, istringstream, ostringstream 806
sync_with_stdio 807
tellg and tellp 808
unsetf 808
width 808
write 809

33 The STL Container Classes 811
The Container Classes 812
bitset 814
deque 816
list 819
map 822
multimap 824
multiset 827
queue 829
priority_queue 830
set 831
stack 833
vector 834

34 The STL Algorithms 839
adjacent_find 840
binary_search 840
copy 841

copy_backward 841
count 841
count_if 842
equal 842
equal_range 842
fill and fill_n 843
find 843
find_end 843
find_first_of 843
find_if 844
for_each 844
generate and generate_n 844
includes 845
inplace_merge 845
iter_swap 845
lexicographical_compare 846
lower_bound 846
make_heap 846
max 847
max_element 847
merge 847
min 848
min_element 848
mismatch 848
next_permutation 849
nth_element 849
partial_sort 849
partial_sort_copy 850
partition 850
pop_heap 850
prev_permutation 851
push_heap 851
random_shuffle 851
remove, remove_if, remove_copy, and remove_copy_if 852
replace, replace_copy, replace_if, and replace_copy_if 852
reverse and reverse_copy 853
rotate and rotate_copy 853
search 854
search_n 854
set_difference 854
set_intersection 855
set_symmetric_difference 855
set_union 856
sort 856
sort_heap 856

stable_partition 857
stable_sort 857
swap 857
swap_ranges 858
transform 858
unique and unique_copy 858
upper_bound 859

35 STL Iterators, Allocators, and Function Objects 861
Iterators 862
The Basic Iterator Types 862
The Low-Level Iterator Classes 863
The Predefined Iterators 864
Two Iterator Functions 872
Function Objects 872
Function Objects 873
Binders 874
Negators 875
Adaptors 876
Allocators 879

36 The String Class 881
The basic_string Class 882
The char_traits Class 894

37 The Numeric Classes 897
The complex Class 898
The valarray Class 902
The slice and gslice Classes 917
The Helper Classes 920
The Numeric Algorithms 920
accumulate 920
adjacent_difference 921
inner_product 922
partial_sum 923

38 Exception Handling and Miscellaneous Classes 925
Exceptions 926
<exception> 926
<stdexcept> 927
auto_ptr 928
The pair Class 930
Localization 931
Other Classes of Interest 931

Part V
Applying C++

39 Integrating New Classes: A Custom String Class 935
The StrType Class 936
The Constructors and Destructors 938
I/O on Strings 939
The Assignment Functions 941
Concatenation 942
Substring Subtraction 945
The Relational Operators 947
Miscellaneous String Functions 948
The Entire StrType Class 949
Using the StrType Class 958
Creating and Integrating New Types in General 961
A Challenge 961

40 Parsing Expressions 963
Expressions 964
Parsing Expressions: The Problem 965
Parsing an Expression 966
The Parser Class 968
Dissecting an Expression 969
A Simple Expression Parser 971
    Understanding the Parser 977
Adding Variables to the Parser 978
Syntax Checking in a Recursive-Descent Parser 988
Building a Generic Parser 989
Some Things to Try 997

A The .NET Managed Extensions to C++ 999
The .NET Keyword Extensions 1000
Preprocessor Extensions 1002
The attribute Attribute 1003
Compiling Managed C++ 1003

B C++ and the Robotics Age 1005

Index 1009