

3.2	Component Interchangeability	46
3.3	Algorithm/Container Compatibility	48
4	Iterators	49
4.1	Input Iterators	50
4.2	Output Iterators	52
4.3	Forward Iterators	54
4.4	Bidirectional Iterators	55
4.5	Random Access Iterators	56
4.6	The STL Iterator Hierarchy: Combining Algorithms and Containers Efficiently	58
4.7	Insert Iterators	59
4.8	Revisiting Input and Output: Stream Iterators	62
4.9	Specification of Iterator Categories Required by STL Algorithms	64
4.10	Designing Generic Algorithms	65
4.11	Why Some Algorithms Require More Powerful Iterators	67
4.12	Choosing the Right Algorithm	68
4.13	Constant Versus Mutable Iterator Types	68
4.14	Iterator Categories Provided by STL Containers	71
5	Generic Algorithms	73
5.1	Basic Algorithm Organization in STL	73
5.2	Nonmutating Sequence Algorithms	77
5.3	Mutating Sequence Algorithms	87
5.4	Sorting-Related Algorithms	102
5.5	Generalized Numeric Algorithms	122
6	Sequence Containers	127
6.1	Vectors	129
6.2	Dequeues	148
6.3	Lists	152
7	Sorted Associative Containers	161
7.1	Sets and Multisets	162
7.2	Maps and Multimaps	174
8	Function Objects	183
8.1	Passing Functions via Function Pointers	184

8.2	Advantages of Specifying Function Objects with Template Parameters	186
8.3	STL-Provided Function Objects	191
9	Container Adaptors	193
9.1	Stack Container Adaptor	194
9.2	Queue Container Adaptor	196
9.3	Priority Queue Container Adaptor	198
10	Iterator Adaptors	201
11	Function Adaptors	205
11.1	Binders	205
11.2	Negators	206
11.3	Adaptors for Pointers to Functions	208
II	Putting It Together: Example Programs	213
12	Program for Searching a Dictionary	215
12.1	Finding Anagrams of a Given Word	215
12.2	Interacting with the Standard String and I/O Streams Classes	218
12.3	Generating Permutations and Searching the Dictionary . .	220
12.4	Complete Program	221
12.5	How Fast Is It?	223
13	Program for Finding All Anagram Groups	225
13.1	Finding Anagram Groups	225
13.2	Defining a Data Structure to Work with STL	226
13.3	Creating Function Objects for Comparisons	227
13.4	Complete Anagram Group Finding Program	228
13.5	Reading the Dictionary into a Vector of PS Objects	229
13.6	Using a Comparison Object to Sort Word Pairs	230
13.7	Using an Equality Predicate Object to Search for Adjacent Equal Elements	230
13.8	Using a Function Adaptor to Obtain a Predicate Object . .	231
13.9	Copying the Anagram Group to the Output Stream	232
13.10	Output of the Anagram Program	232

14 Better Anagram Program: Using the List and Map Containers	235
14.1 Data Structure Holding Iterator Pairs	235
14.2 Storing Information in a Map of Lists	236
14.3 Outputting the Anagram Groups in Order of Size	237
14.4 Better Anagram Program	238
14.5 Output of the Program	239
14.6 Why Use a Map Container?	240
15 Faster Anagram Program: Using Multimaps	243
15.1 Finding Anagram Groups, Version 3	243
15.2 Declaration of the Multimap	246
15.3 Reading the Dictionary into the Multimap	247
15.4 Finding the Anagram Groups in the Multimap	247
15.5 Outputting the Anagram Groups in Order of Size	249
15.6 Output of the Program	249
15.7 How Fast Is It?	250
16 Defining an Iterator Class	251
16.1 New Kind of Iterator: Counting Iterator	251
16.2 Counting Iterator Class	253
17 Combining STL with Object-Oriented Programming	259
17.1 Using Inheritance and Virtual Functions	260
17.2 Avoiding “Code Bloat” from Container Instances	265
18 Program for Displaying Theoretical Computer Science Genealogy	267
18.1 Sorting Students by Date	267
18.2 Associating Students with Advisors	268
18.3 Finding the Roots of the Tree	270
18.4 Reading the File	273
18.5 Printing the Results	275
18.6 Complete “Genealogy” Program	276
19 Class for Timing Generic Algorithms	279
19.1 Obstacles to Accurate Timing of Algorithms	279
19.2 Overcoming the Obstacles	280
19.3 Refining the Approach	283
19.4 Automated Analysis with a Timer Class	284

19.5	Timing the STL Sort Algorithms	289
III STL Reference Guide		293
20	Iterator Reference Guide	295
20.1	Input Iterator Requirements	296
20.2	Output Iterator Requirements	298
20.3	Forward Iterator Requirements	299
20.4	Bidirectional Iterator Requirements	299
20.5	Random Access Iterator Requirements	300
20.6	Iterator Traits	301
20.7	Iterator Operations	304
20.8	Istream Iterators	304
20.9	Ostream Iterators	307
20.10	Reverse Iterators	309
20.11	Back Insert Iterators	313
20.12	Front Insert Iterators	315
20.13	Insert Iterators	316
21	Container Reference Guide	319
21.1	Requirements	319
21.2	Organization of the Container Class Descriptions	330
21.3	Vector	331
21.4	Deque	339
21.5	List	345
21.6	Set	354
21.7	Multiset	360
21.8	Map	365
21.9	Multimap	373
21.10	Stack Container Adaptor	378
21.11	Queue Container Adaptor	380
21.12	Priority Queue Container Adaptor	383
22	Generic Algorithm Reference Guide	387
22.1	Organization of the Algorithm Descriptions	387
22.2	Nonmutating Sequence Algorithm Overview	389
22.3	For Each	390
22.4	Find	391
22.5	Find First	391

22.6	Adjacent Find	392
22.7	Count	393
22.8	Mismatch	394
22.9	Equal	395
22.10	Search	396
22.11	Search N	397
22.12	Find End	397
22.13	Mutating Sequence Algorithm Overview	398
22.14	Copy	399
22.15	Swap	400
22.16	Transform	401
22.17	Replace	402
22.18	Fill	403
22.19	Generate	404
22.20	Remove	404
22.21	Unique	406
22.22	Reverse	407
22.23	Rotate	408
22.24	Random Shuffle	408
22.25	Partition	409
22.26	Sorting-Related Algorithms Overview	410
22.27	Sort	412
22.28	Nth Element	414
22.29	Binary Search	415
22.30	Merge	417
22.31	Set Operations on Sorted Structures	418
22.32	Heap Operations	421
22.33	Min and Max	423
22.34	Lexicographical Comparison	424
22.35	Permutation Generators	425
22.36	Generalized Numeric Algorithms Overview	426
22.37	Accumulate	427
22.38	Inner Product	428
22.39	Partial Sum	429
22.40	Adjacent Difference	430
23	Function Object and Function Adaptor Reference Guide	431
23.1	Requirements	431
23.2	Base Classes	432

23.3	Arithmetic Operations	433
23.4	Comparison Operations	433
23.5	Logical Operations	434
23.6	Negator Adaptors	435
23.7	Binder Adaptors	435
23.8	Adaptors for Pointers to Functions	436
23.9	Adaptors for Pointers to Member Functions	437
24	Allocator Reference Guide	441
24.1	Introduction	441
24.2	Allocator Requirements	442
24.3	Default Allocator	445
24.4	Custom Allocators	448
25	Utilities Reference Guide	455
25.1	Introduction	455
25.2	Comparison Functions	455
25.3	Pairs	456
Appendix A	STL Header Files	459
Appendix B	String Reference Guide	461
B.1	String Classes	461
B.2	Character Traits	472
Appendix C	STL Include Files Used in Example Programs	477
C.1	Files Used in Example ??	477
Appendix D	STL Resources	483
D.1	Internet Addresses for SGI Reference Implementation of STL	483
D.2	World Wide Web Address for Source Code for Examples in this Book	483
D.3	STL-Compatible Compilers	484
D.4	Other Related STL and C++ Documents	484
D.5	Generic Programming and STL Discussion List	484
References		485
Index		485