## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreword</td>
<td>xv</td>
</tr>
<tr>
<td>Preface</td>
<td>xvii</td>
</tr>
<tr>
<td>Guide for Readers</td>
<td>xxi</td>
</tr>
<tr>
<td><strong>PART I: STREAM INPUT AND OUTPUT</strong></td>
<td></td>
</tr>
<tr>
<td>1. IOStreams Basics</td>
<td></td>
</tr>
<tr>
<td>1.1 Input and Output</td>
<td></td>
</tr>
<tr>
<td>1.2 Formatted Input/Output</td>
<td></td>
</tr>
<tr>
<td>1.2.1 The Predefined Global Streams</td>
<td></td>
</tr>
<tr>
<td>1.2.2 The Input and Output Operators</td>
<td></td>
</tr>
<tr>
<td>1.2.3 The Format Parameters of a Stream</td>
<td></td>
</tr>
<tr>
<td>1.2.4 Manipulators</td>
<td></td>
</tr>
<tr>
<td>1.2.5 The Locale of a Stream</td>
<td></td>
</tr>
<tr>
<td>1.2.6 Comparison Between Formatted Input and Output</td>
<td></td>
</tr>
<tr>
<td>1.2.7 Peculiarities of Formatted Input</td>
<td></td>
</tr>
<tr>
<td>1.3 The Stream State</td>
<td></td>
</tr>
<tr>
<td>1.3.1 The Stream State Flags</td>
<td></td>
</tr>
<tr>
<td>1.3.2 Checking the Stream State</td>
<td></td>
</tr>
<tr>
<td>1.3.3 Catching Stream Exceptions</td>
<td></td>
</tr>
<tr>
<td>1.3.4 Resetting the Stream State</td>
<td></td>
</tr>
<tr>
<td>1.4 File Input/Output</td>
<td></td>
</tr>
<tr>
<td>1.4.1 Creating, Opening, Closing and Destroying File Streams</td>
<td></td>
</tr>
<tr>
<td>1.4.2 The Open Modes</td>
<td></td>
</tr>
<tr>
<td>1.4.3 Bidirectional File Streams</td>
<td></td>
</tr>
</tbody>
</table>
1.5 In-Memory Input/Output
1.6 Unformatted Input/Output
1.7 Stream Positioning
1.8 Synchronization of Streams
  1.8.1 Means of Synchronization
    1.8.1.1 Synchronization via flush() and sync()
    1.8.1.2 Synchronization using the unitbuf Format Flag
    1.8.1.3 Synchronization by Tying Streams
  1.8.2 Synchronizing the Predefined Standard Streams
    1.8.2.1 Synchronization Among the Predefined Standard Streams
    1.8.2.2 Synchronization with the C Standard I/O
    1.8.2.3 Synchronization with the External Device
    1.8.2.4 Synchronization Between Predefined Standard Streams
      for Narrow and Wide Characters
2. The Architecture of IOStreams
2.1 The Stream Classes
  2.1.1 Class Hierarchy
    2.1.1.1 The Stream Base Classes
    2.1.1.2 The General Stream Classes
    2.1.1.3 The Concrete Stream Classes
  2.1.2 How Streams Maintain Their Stream Buffer
  2.1.3 Copying and Assignment of Streams
  2.1.4 How Streams Maintain Their Locale
  2.1.5 Collaboration Among Streams, Stream Buffers, and Locales
2.2 The Stream Buffer Classes
  2.2.1 Class Hierarchy
  2.2.2 The Stream Buffer Abstraction
  2.2.3 String Stream Buffers
  2.2.4 File Stream Buffers
2.3 Character Types and Character Traits
  2.3.1 Character Representations
  2.3.2 Character Traits
    2.3.2.1 Requirements of a Character Traits Type
    2.3.2.2 The Predefined Standard Character Traits
  2.3.3 Character Types
    2.3.3.1 Requirements for Character Types
2.4 Stream Iterators and Stream Buffer Iterators
  2.4.1 The Concepts of Iterators in the Standard Library
  2.4.2 Stream Iterators
    2.4.2.1 Output Stream Iterator
    2.4.2.2 Input Stream Iterator
3.3.2 Creating New Stream Classes by Derivation
  3.3.2.1 Deriving the New Stream Class
  3.3.2.2 Implementing the Date Inserter and the Manipulator
  3.3.2.3 Summary
3.3.3 Comparing Both Solutions—iword/pword versus Derivation
3.4 Adding Stream Buffer Functionality
  3.4.1 Deriving from the Stream Buffer Base Class
    3.4.1.1 Core Functionality of Stream Buffers: Character Transportation
    3.4.1.2 Optional Functionality of Stream Buffers
    3.4.1.3 Providing New Stream Classes Along with New Stream Buffer Classes
  3.4.2 Deriving from Concrete Stream Buffer Classes

PART II: INTERNATIONALIZATION

4. Introduction to Internationalization and Localization
  4.1 Internationalization and Localization
  4.2 Cultural Conventions
    4.2.1 Language
    4.2.2 Numerical Values
    4.2.3 Monetary Values
    4.2.4 Time and Date
    4.2.5 Sorting Words
    4.2.6 Messages
    4.2.7 Character Encodings
      4.2.7.1 Terms and Definitions
      4.2.7.2 Character Codesets
      4.2.7.3 Character Encoding Schemes
      4.2.7.4 Uses of Multibyte Encodings and Wide Characters
      4.2.7.5 Code Conversions
  5. Locales
    5.1 Creating Locale Objects
      5.1.1 Named Locales
      5.1.2 Combined Locales
      5.1.3 The Global Locale
    5.2 Retrieving Facets from a Locale
      5.2.1 has_facet
      5.2.2 use_facet
  6. Standard Facets
    6.1 Alphabet- and Language-Related Facets
      6.1.1 Character Classification
        6.1.1.1 Character Classification
8. User-Defined Facets

8.1 Adding a User-Defined Facet to an Existing Facet Family

8.2 Defining a New Facet Family

### Reference Guide

**Introduction**

1. Locale
   - header file `<locale>`
   - global functions
   - `codecvt<internT,externT,stateT>`
   - `codecvt_base`
   - `codecvt_byname<internT,externT,stateT>`
   - `collate<charT>`
   - `collate_byname<charT>`
   - `ctype<charT>`
   - `ctype<char>`
   - `ctype_base`
   - `ctype_byname<charT>`
   - `locale`
   - `messages<charT>`
   - `messages_base`
   - `messagesbyname<charT>`
   - `money_base`
   - `money_get<charT,InputIterator>`
   - `moneypunct<charT,Inter>`
   - `moneypunctbyname<charT,Inter>`
   - `money_put<charT,OutputIterator>`
   - `num_get<charT,InputIterator>`
   - `numpunct<charT>`
   - `numpunctbyname<charT>`
   - `num_put<charT,OutputIterator>`
   - `time_base`
   - `time_get<charT,InputIterator>`
   - `time_getbyname<charT,InputIterator>`
   - `time_put<charT,OutputIterator>`
   - `time_putbyname<charT,OutputIterator>`
2. Character Traits 440
   header file<string>
   char_traits<charT>
   char_traits<char>
   char_traits<wchar_t>

3. IOStreams 451
   header file<iosfwd>
   global type definitions
   global objects
   basic_filebuf<charT,traits>
   basic_fstream<charT,traits>
   basic_ifstream<charT,traits>
   basic_ios<charT,traits>
   basic_iostream<charT,traits>
   basic_istream<charT,traits>
   basic_istringstream<charT,traits,Allocator>
   basic_ofstream<charT,traits>
   basic_ostream<charT,traits>
   basic_ostringstream<charT,traits,Allocator>
   basic_streambuf<charT,traits>
   basic_stringbuf<charT,traits,Allocator>
   basic_stringstream<charT,traits,Allocator>
   fpos<stateT>
   ios_base
   manipulators

4. Stream Iterators 540
   header file<iterator>
   istreambuf_iterator<charT,traits>
   istream_iterator<T,charT,traits,Distance>
   iterator<Category,T,Distance,Pointer,Reference>
   iterator category tags
   ostreambuf_iterator<charT,traits>
   ostream_iterator<T,charT,traits,Distance>
5. Other I/O Operations

bitset\langle N\rangle
complex\langle T\rangle
basic_string\langle charT,traits,Allocator\rangle

APPENDICES

Appendix A: Parsing and Extraction of Numerical and bool Values
A.1 Parsing Numerical Values
A.2 Parsing bool Values
A.3 Conversion Specifier and Length Modifier

Appendix B: Formatting of Numerical and bool Values
B.1 Formatting Numerical Values
B.2 Formatting bool Values
B.3 Conversion Specifiers, Qualifiers, and Length Modifiers

Appendix C: strftime() Conversion Specifiers

Appendix D: Correspondences Between C Stdio and C++IOStreams
D.1 File Open Modes
D.2 Stream Positions

Appendix E: Differences Between Classic and Standard IOStreams
E.1 Templatizing the IOStreams Classes
E.2 Splitting Base Class ios
E.3 Indicating Errors
E.4 Internationalizing IOStreams
E.5 Removing _withassign Classes
E.6 Removing File Descriptors
E.7 String Streams: Replacing strstream by stringstream
E.8 Changes to the Stream Buffer Classes
E.9 Minor Changes

Appendix F: The Relationship Between C and C++ Locales
F.1 Locale Categories in C and C++
F.2 The Global Locale in C and C++

Appendix G: New C++ Features and Idioms
G.1 Bitmask Types
G.2 POD—Plain Old Data