Contents

1. OVERVIEW 1

1.1 Assumptions 2
1.2 What Is Covered 2
1.3 System Requirements 3
1.4 C# History 4
1.5 Languages and Standards 5
1.6 Algorithms and Libraries 6
1.7 News Groups 6
1.8 Bibliography 7

2. LANGUAGE CONCEPTS 9

2.1 Programming Language Developments 10
2.1.1 Compilation Style 10
2.1.2 Data Types 11
2.1.3 Dynamic Arrays 11
2.1.4 Polymorphism 11
2.1.5 Structured Programming 12
2.1.6 Stepwise Refinement 12
2.1.7 Information Hiding and Modules 12
2.1.8 Pointers and References 13
2.1.9 Operator Overloading 13
2.1.10 Threads and Multitasking 13
2.1.11 Exception Handling 14
2.1.12 Object-Oriented Programming 14
2.1.12.1 The Approach of Meyer 15
2.1.12.2 The Approach of Rumbaugh, et al. 16
2.1.13 Development Environments 17
2.1.14 Graphical Development Tools 17
2.1.15 Software Components 18
2.2 C# Concepts and Terminology 19
2.2.1 Classes 19
2.2.1.1 Constants 20
2.2.1.2 Fields 20
2.2.1.3 Methods 20
2.2.1.4 Properties 20
2.2.1.5 Events 20
2.2.1.6 Indexers 21

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page vi

2.2.1.7 Instance Constructors 21
2.2.1.8 Static Constructors 21
2.2.1.9 Destructors 21
2.2.2 Inheritance 21
2.2.3 Interfaces 21
2.2.4 Delegates 21
2.2.5 Attributes 22
2.3 Bibliography 22
2.3.1 Books and Articles 22
2.3.2 Web Addresses 26
2.3.2.1 Search engines 27
2.3.2.2 Ada 27
2.3.2.3 C 27
2.3.2.4 C++ 27
2.3.2.5 C# 27
2.3.2.6 Eiffel 27
2.3.2.7 Fortran 27
2.3.2.8 Microsoft Visual Basic 27
2.3.2.9 Modula 2 27
2.3.2.10 Oberon 27
2.3.2.11 Pascal 27

3. AN INTRODUCTION TO PROGRAMMING IN C# 29
3.1 Introduction 30
3.2 Hello World Using the Console 32
3.2.1 using System; 32
3.2.2 class HelloWorld 32
3.2.3 { and } 32
3.2.4 static int Main() 32
3.2.5 Console.WriteLine("Hello World"); 32
3.2.6 return 0; 33
3.2.7 ; 33
3.3 Hello World Using Developer Studio 33
3.3.1 Starting Developer Studio 33
3.3.2 Creating a New Project 35
3.3.3 Creating an Empty Project 36
3.3.4 Adding an Existing Item 37
3.3.5 Building the Project 39
3.3.6 Running the Project 40
3.3.7 Editing the Project 42
3.4 Console Application 42

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page vii

3.5 Simple Text I/O 44
3.6 Simple Numeric I/O 44
3.7 On-Line Documentation 47
3.7.1 Contents Mode 49
3.7.2 Compiler Error Messages 52
3.8 Key Concepts of C# 53
3.9 Summary 53

4. ARITHMETIC AND EXPRESSIONS 55
4.1 Introduction 56
4.2 Integral Types 58
4.3 Floating Point Types 60
4.4 The Decimal Type 62
4.5 Math Members 63
4.5.1 Public Static (Shared) Fields 63
4.5.2 Public Static (Shared) Methods 63
4.5.3 Public Instance Methods 64
4.5.4 Protected Instance Methods 64
4.6 Operators 65
4.7 Simple and Aliased Types 69
4.8 Integer Arithmetic 70
4.8.1 Integer Overflow 70
4.8.2 Evaluating Simple Expressions 72
4.8.3 Evaluating Expressions Using Increment Operators 73
4.8.4 Evaluating Expressions Using Integer Division and Remainder 73
4.8.5 Variation on a Theme 74
4.9 Floating Point Arithmetic 75
4.9.1 Simple Single and Double Precision Example 75
4.9.2 Comparing Calculations 75
4.9.2.1 Using Variables of Type float 76
4.9.2.2 Using Variables of Type double 77
4.9.3 Expression Equivalence 78
4.9.4 Trigonometric Evaluation 79
4.9.5 Base Conversion 82
4.9.6 Overflow and Underflow 82
4.9.7 Using an Uninitialised Variable 83
4.10 Formatting Numbers 84
4.11 Escape Sequences 85
4.12 Literals 85
4.13 Arithmetic in Other languages 86
4.14 Summary 86

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page viii

4.15 Bibliography 87
4.16 Problems 87

5. SOME MORE ON DATA TYPES 89
5.1 Booleans 90
5.2 Characters and Strings 90
5.2.1 The char Data Type 92
5.2.2 Arrays of Characters 92
5.2.3 Initialization and Object Assignment 93
5.2.4 Accessing Array Elements 94
5.3 Value and Reference Types 94
5.4 Enum Types 95
5.4.1 Days of the Week Example 95
5.4.2 Colours Example 96
5.5 Other Types 97
5.5.1 Struct Types 97
5.5.2 Class Types 97
5.5.3 Object Types 97
5.5.4 Array Types 98
5.6 C# Keywords 98
5.7 Comments 98
5.8 Summary 100
5.9 Problem 100

6. STRINGS 101
6.1 Basic String Manipulation 102
6.2 Concatenating and Splitting Strings 103
6.3 Using the Length Property and IndexOf Method 105
6.4 String Constructors, Fields, Methods, Operators and Properties 106
6.5 Problems 106

7. ARRAYS 107
7.1 Introduction 108
7.2 One-dimensional Arrays 108
7.2.1 Using a for Statement 108
7.2.2 Using a foreach Statement 110
7.3 Array Initialization 110
7.4 Simple Two-dimensional Arrays 111
7.5 Arrays of Arrays 113
7.6 Ragged Arrays 114

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page ix

7.7 List of Array Members 115
7.7.1 Public Static (Shared) Methods 115
7.7.2 Public Instance Properties 115
7.7.3 Public Instance Methods 116
7.7.4 Protected Instance Constructors 116
7.7.5 Protected Instance Methods 116
7.8 Summary 117
7.9 Problems 117

8. CONTROL STRUCTURES 119
8.1 Statement End Points and Reachability 120
8.2 Types of Statements 121
8.2.1 Labeled Statements 121
8.2.2 Declaration Statements 121
8.2.3 Embedded Statements 122
8.2.3.1 Blocks 122
8.2.3.2 Empty Statements 122
8.2.3.3 Expression Statements 122
8.2.3.4 Checked and Unchecked Statements 123
8.2.3.5 Lock Statements 123
8.2.3.6 Using Statements 123
8.3 Selection Statements 124
8.3.1 If Statement 124
8.3.2 Switch Statement 125
8.4 Iteration Statements 130
8.4.1 While Statement 130
8.4.2 Do Statement 131
8.4.3 For Statement 133
8.4.4 Foreach Statement 134
8.5 Jump Statements 135
8.5.1 Break Statement 135
8.5.2 Continue Statement 136
8.5.3 Goto Statement 136
8.5.4 Return Statement 136
8.5.5 Example of Jump Statements 137
8.6 Exception Handling Using Try, Catch Throw 138
8.7 Summary 139
8.8 Bibliography 139
8.9 Problems 139

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page x

9. INPUT AND OUTPUT 143
9.1 Introduction 144
9.2 Using the Console Class 145
9.3 Using the StreamReader Class 146
9.4 Specifying the File Location 147
9.5 Reading from Standard Input 148
9.6 Reading from the Command Line 149
9.7 Using StreamReader and StreamWriter with Hard-coded File Names 150
9.8 Reading a File from a Web Server 151
9.9 Reading Tabular Data from a Text File 152
9.10 The System.Console Class 153
9.11 The StreamReader Class 154
9.12 The System.IO.File Class 155
9.13 Problems 157

10. EXCEPTIONS 161
10.1 Introduction 162
10.2 The Exception Hierarchy 163
10.2.1 Single Public Fields 163
10.2.2 Single Public Methods 164
10.3 Integer Arithmetic Exceptions 164
10.3.1 Default Run Time Behaviour 164
10.3.2 Checked Behaviour 165
10.4 Floating Point Exceptions 166
10.4.1 Default Run Time Behaviour 166
10.4.2 Checked Behaviour 167
10.4.3 Throwing an Exception 167
10.5 Summary 168

11. THREADS 169
11.1 Introduction 170
11.2 Basic Thread Syntax 171
11.3 Using the Sleep Method 172
11.4 Threads and Static Variables 173
11.5 Threads with Start, Suspend and Resume 175
11.6 Thread Priority 176
11.7 Thread States 179
11.8 Microsoft Examples 179
11.9 Bibliography 182
11.10 Problems 183

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page xi

12. STRUCTS 185
12.1 Introduction 186
12.2 Basic Syntax 186
12.3 Simple Example 187

13. CLASSES 189
13.1 Basic Class Syntax 190
13.2 Examples 191
13.2.1 Greatest Common Divider 191
13.2.2 Recursion 192
13.2.3 Checked Arithmetic 193
13.2.4 Using Try and Catch 194
13.2.5 Date Class 196
13.2.6 Arrays of Dates 197
13.2.7 Inheritance: Virtual Date Base Class 198
13.3 Passing Parameters 201
13.4 Instance and Static Members 202
13.4.1 Compiler Error CS0120 205
13.4.2 Compiler Error CS0176 206
13.5 Additional Syntax 207
13.6 Bibliography 208

14. THE SYSTEM.COLLECTIONS NAMESPACE 209
14.1 Introduction 210
14.2 Using the Hashtable Class 211
14.3 Using the SortedList Class 214
14.4 A Spelling Checker 216
14.5 Summary 219
14.6 Bibliography 219

15. OPERATOR OVERLOADING 221
15.1 Complex Numbers 222
15.2 Complex Arithmetic in Other Languages 222
15.3 Complex Numbers Using the Float Data Type 223
15.4 Complex Numbers Using the Double Data Type 226
15.5 Complex Arithmetic Using Implicit Type Conversions 227
15.6 Complex Arithmetic Using Explicit Type Conversions 229
15.7 Arrays of Complex Numbers 231
15.8 Summary 232

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page xii

16. AN INTRODUCTION TO WINDOWS PROGRAMMING 233

16.1 Introduction 234
16.2 The .Net IDE 234
16.2.1 Other Languages and the .Net Environment 235
16.2.2 Visual Development 235
16.3 Buttons and Event Handling Using a MessageBox Control 245
16.3.1 Getting Help when Writing Programs 248
16.3.2 Running the Program 251
16.4 Using a TextBox to Accept User Input 256
16.5 Using a RichTextBox to Display Program Output 258
16.6 Using a PictureBox to Display an Image 260
16.7 Selecting a File with the OpenFileDialog Control 261
16.8 Text and Graphics Using GDI+ 264
16.8.1 GDI+ 264
16.8.2 Creating Graphical Images with GDI+ 265
16.8.3 Rendering with GDI+ 266
16.8.4 Coordinate Systems 267
16.8.5 The OnPaint Method 267
16.8.6 Graphics Objects 267
16.8.7 Simple Example Using the DrawLine Method 268
16.8.8 Default Coordinate System: Plotting Sines and Cosines 271
16.8.9 Alternate Coordinate System: Plotting Sines and Cosines 275
16.8.10 Simple Text Example 277
16.9 Converting a Console to a Windows Application 279
16.10 Bibliography 283

17. ADDITIONAL LANGUAGE FEATURES 285

17.1 Events and Event Handling 286
17.2 Interfaces 287
17.3 Attributes 287
17.4 Delegates 287
17.5 Windows-based Programming 288
17.6 Namespaces 289
17.7 Assemblies and Versioning 289
17.8 Bibliography 290

18. WHERE NEXT?. 291

18.1 Online Documentation 292
18.2 Examples 292
18.3 Tutorials 292
18.4 Wizards 292

3613 prelims PT REVISE/gk 14/2/03 4:26 pm Page xiii

18.5 Web Sources 293
18.6 Books 293
18.7 Microsoft 293
18.8 Source Code 293
18.9 Coda 293

APPENDIX A
Default Coordinate System Example in C++ 294

INDEX 297



http://www.springer.com/978-1-85233-562-5

Essential C# fast
chivers, I.
2003, XIII, 303 p., Softcover
ISBN: 978-1-85233-562-5