

INTERNATIONAL EDITION

THIRD EDITION

AN OBJECT-ORIENTED
APPROACH TO
PROGRAMMING
LOGIC AND DESIGN
JOYCE FARRELL



Not for Sale in the
United States

Contents

	Preface	xiv
CHAPTER 1	An Overview of Computer Programming	1
	Understanding Computer Components and Operations	2
	Understanding Simple Program Logic	4
	Understanding the Evolution of Programming Techniques	6
	Understanding the Steps in the Programming Process.	8
	Analyzing the System.	8
	Designing the System	9
	Writing the Program	10
	Using Pseudocode and Flowcharts	15
	Writing Pseudocode	15
	Drawing Flowcharts	16
	Understanding Program Comments	19
	Understanding Programming and User Environments	20
	Understanding Programming Environments	20
	Understanding User Environments	23
	Chapter Summary	24
	Key Terms	24
	Review Questions	28
	Exercises	33
	Case Problems	33
	Up for Discussion	34
CHAPTER 2	Working with Classes, the Main Method, and Data	35
	Creating an Application Class with a <code>main()</code> Method	36
	Understanding the <code>main()</code> Method	38
	Understanding How Programming Languages Reflect Logic	39

Using Literals, Variables, and Named Constants	39
Understanding Unnamed, Literal Constants and their Data Types	40
Working with Variables	40
Naming Variables	42
Understanding a Variable's Data Type	42
Declaring Named Constants.	43
Assigning Values to Variables	44
Performing Arithmetic Operations	45
Features of Good Program Design.	48
Using Program Comments	48
Choosing Identifiers	49
Designing Clear Statements.	51
Writing Clear Prompts and Echoing Input.	52
Maintaining Good Programming Habits	54
An Introduction to Structure and Creating a Complete Program	55
Chapter Summary	58
Key Terms	59
Review Questions	62
Exercises	66
Case Problems	68
Up for Discussion	69

CHAPTER 3	Making Decisions	70
	Evaluating Boolean Expressions to Make Comparisons	71
	Using the Relational Comparison Operators	76
	Avoiding a Common Error with Relational Operators.	80
	Understanding AND Logic	80
	Nesting AND Decisions for Efficiency	82
	Using the AND Operator	85
	Avoiding Common Errors in an AND Selection.	87
	Understanding OR Logic	88
	Writing OR Decisions for Efficiency	90
	Using the OR Operator	92
	Avoiding Common Errors in an OR Selection	93
	Making Selections within Ranges	98
	Avoiding Common Errors When Using Range Checks	101
	Understanding Precedence When Combining AND and OR Operators	105
	Chapter Summary	108
	Key Terms	109
	Review Questions	110

CHAPTER 4

Exercises	116
Case Problems	120
Up for Discussion	123

Looping 124

Understanding Loop Basics	125
Using a Loop Control Variable	126
Using a Definite Loop with a Counter	127
Using an Indefinite Loop with a Sentinel Value	128
Using Nested Loops	130
Avoiding Common Loop Mistakes	133
Mistake: Neglecting to Initialize the Loop Control Variable	133
Mistake: Neglecting to Alter the Loop Control Variable	135
Mistake: Using the Wrong Comparison with the Loop Control Variable	137
Mistake: Including Statements Inside the Loop that Belong Outside the Loop	138
Using a For Loop	142
Common Loop Applications	144
Using a Loop to Accumulate Totals	145
Using a Loop to Validate Data	148
Limiting a Reprompting Loop	150
Validating a Data Type	152
Validating Reasonableness and Consistency of Data	153
Chapter Summary	154
Key Terms	154
Review Questions	156
Exercises	160
Case Problems	163
Up for Discussion	166

CHAPTER 5

Arrays 167

Understanding Arrays and How They Occupy Computer Memory	168
How Arrays Occupy Computer Memory	169
Manipulating an Array to Replace Nested Decisions	171
Using Constants with Arrays	179
Using a Named Constant as the Size of an Array	180
Using Constants as Array Element Values	180
Using a Constant as an Array Subscript	180
Searching an Array	181

Using Parallel Arrays	184
Improving Search Efficiency.	187
Searching an Array for a Range Match	188
Remaining within Array Bounds	192
Using a for Loop to Process Arrays	194
Chapter Summary	195
Key Terms	196
Review Questions	197
Exercises	201
Case Problems	208
Up for Discussion	212

CHAPTER 6 Using Methods **213**

Understanding the Advantages of Modularization	214
Modularization Provides Abstraction	214
Modularization Simplifies the Logic	215
Modularization Allows Multiple Programmers to Work on a Problem	215
Modularization Allows You to Reuse Your Work	216
Modularizing a Program	217
Declaring Variables and Constants within Methods	221
Creating Methods that Require a Single Parameter	222
Understanding the Difference Between Passing Arguments by Value and by Reference	226
Creating Methods that Require Multiple Parameters	228
Creating Methods that Return a Value	230
Passing an Array to a Method.	235
Overloading Methods.	239
Avoiding Ambiguous Methods	242
Using Predefined Methods	245
Chapter Summary	246
Key Terms	248
Review Questions	250
Exercises	254
Case Problems	257
Up for Discussion	261

CHAPTER 7 Object-Oriented Programming Concepts . . . **262**

Principles of Object-Oriented Programming.	263
Classes and Objects	263
Polymorphism	266

Inheritance	267
Encapsulation	268
Defining Classes and Creating Class Diagrams	268
Creating Class Diagrams	271
The Set Methods.	274
The Get Methods.	275
Work Methods	276
Understanding Public and Private Access	277
Organizing Classes	281
Understanding Instance Methods	282
Understanding Static, Class Methods	287
Using Objects	288
Understanding Composition.	292
One Example of Using Predefined Classes:	
Creating GUI Objects	293
Reviewing the Advantages of Object-Oriented	
Programming	294
Chapter Summary.	294
Key Terms	296
Review Questions	298
Exercises	302
Case Problems	305
Up for Discussion	310

CHAPTER 8	More Object Concepts	311
	An Introduction to Constructors	312
	Constructors with Parameters.	315
	Overloading Instance Methods and Constructors	316
	Understanding Destructors	317
	Understanding Inheritance	319
	Understanding Inheritance Terminology	322
	Accessing Private Members of a Parent Class	325
	Overriding Base Class Methods	330
	Understanding How Constructors Are Called During	
	Inheritance	334
	Understanding How a Derived Class Object "Is An"	
	Instance of the Base Class	337
	Using Inheritance to Achieve Good Software Design.	337
	Chapter Summary	338
	Key Terms	340
	Review Questions	341
	Exercises	345

Case Problems	348
Up for Discussion	350

CHAPTER 9

Event-Driven Programming with Graphical User Interfaces **351**

Understanding Event-Driven Programming	352
User-Initiated Actions and GUI Components.	354
Designing Graphical User Interfaces	358
The Interface Should Be Natural and Predictable	358
The Interface Should Be Attractive, Easy to Read, and Nondistracting	359
To Some Extent, It's Helpful If the User Can Customize Your Applications.	359
The Program Should Be Forgiving	360
The GUI Is Only a Means to an End	360
The Steps to Developing an Event-Driven Application	360
Creating Storyboards.	361
Defining the Storyboard Objects in an Object Dictionary	361
Defining Connections Between the User Screens	363
Planning the Logic	363
Understanding Multithreading	368
Creating Animation.	370
Chapter Summary	373
Key Terms	374
Review Questions	375
Exercises	380
Case Problems	381
Up for Discussion	382

CHAPTER 10

Exception Handling **384**

Learning About Exceptions	385
Understanding the Limitations of Traditional Error Handling	387
Trying Code and Catching Exceptions	388
Throwing and Catching Multiple Exceptions.	391
Using the finally Block	395
Understanding the Advantages of Exception Handling	397
Tracing Exceptions Through the Call Stack	400
A Case Study: Tracing the Source of an Exception	402
Creating Your Own Exceptions	405
Chapter Summary	408
Key Terms	409
Review Questions	410

CHAPTER 11

Exercises	414
Case Problems	417
Up for Discussion	422

System Modeling with the UML 423

Understanding the Need for System Modeling	424
What is the UML?	425
Using Use Case Diagrams	427
Using Class and Object Diagrams	432
Using Sequence and Communication Diagrams	435
Using State Machine Diagrams	437
Using Activity Diagrams	438
Using Component, Deployment, and Profile Diagrams	440
Diagramming Exception Handling	441
Deciding When to Use the UML and Which UML Diagrams to Use	442
Chapter Summary	443
Key Terms	444
Review Questions	446
Exercises	450
Case Problems	452
Up for Discussion	453

CHAPTER 12

Manipulating Larger Quantities of Data . . . 454

Understanding the Need for Sorting Records	455
Understanding How to Swap Two Values	456
Using a Bubble Sort	457
Sorting a List of Variable Size	466
Refining the Bubble Sort to Reduce Unnecessary Comparisons	472
Refining the Bubble Sort to Eliminate Unnecessary Passes	474
Other Sorts	476
Using Multidimensional Arrays	476
Using Indexed Files and Linked Lists	480
Using Indexed Files	481
Using Linked Lists	483
Chapter Summary	485
Key Terms	486
Review Questions	487

	Exercises	492
	Case Problems	496
	Up for Discussion	500
APPENDIX A	Conventions in this Book	501
APPENDIX B	Flowchart Symbols	503
APPENDIX C	Understanding Numbering Systems and Computer Codes.	504
APPENDIX D	Structure	512
	Glossary	522
	Index	536