

GLOBAL
EDITION



Fluency With Information Technology

Skills, Concepts, & Capabilities

SIXTH EDITION

Lawrence Snyder



FLUENCY 6

With Information Technology

SKILLS, CONCEPTS, & CAPABILITIES

Global Edition

LAWRENCE SNYDER

UNIVERSITY OF WASHINGTON

Global Edition contributions by

CHETHAN VENKATESH

M S RAMAIAH INSTITUTE OF TECHNOLOGY

Pearson

Boston Columbus Indianapolis New York San Francisco Upper Saddle River
Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto
Delhi Mexico City Sao Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

Editorial Director, ECS: Marcia Horton
Acquisitions Editor: Matt Goldstein
Director of Marketing: Christy Lesko
Marketing Manager: Yezan Alayan
Marketing Assistant: Jon Bryant
Director of Production: Erin Gregg
Managing Editor: Jeff Holcomb
Senior Project Manager: Marilyn Lloyd
Head, Learning Asset Acquisition, Global Edition:
Laura Dent
Acquisitions Editor, Global Edition: Karthik Subramaniam
Project Editor, Global Edition: Anuprova Dey Chowdhuri

Manufacturing Buyer: Linda Sager
Cover Designer: PreMediaGlobal
Text Designer: Gillian Hall
Manager, Text Permissions: Tim Nicholls
Text Permission Project Manager: Jenell Forschler
Manager, Image Permissions: Karen Sanatar
Cover Image: Shutterstock
Media Project Manager: Renata Butera
Full-Service Project Management: Gillian Hall
Composition: Laserwords
Printer/Binder: Courier Kendallville
Cover Printer: Courier Kendallville

Pearson Education Limited
Edinburgh Gate
Harlow
Essex CM20 2JE
England

and Associated Companies throughout the world

Visit us on the World Wide Web at:
www.pearsonglobaleditions.com

© Pearson Education Limited 2015

The rights of Lawrence Snyder to be identified as the author of this work have been asserted by him in accordance with the Copyright, Designs and Patents Act 1988.

Authorized adaptation from the United States edition, entitled Fluency With Information Technology, 6th edition, ISBN 9780-1-33-57739-6, by Lawrence Snyder, published by Pearson Education © 2015.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without either the prior written permission of the publisher or a license permitting restricted copying in the United Kingdom issued by the Copyright Licensing Agency Ltd, Saffron House, 6–10 Kirby Street, London EC1N 8TS.

All trademarks used herein are the property of their respective owners. The use of any trademark in this text does not vest in the author or publisher any trademark ownership rights in such trademarks, nor does the use of such trademarks imply any affiliation with or endorsement of this book by such owners.

Credits and acknowledgments borrowed from other sources and reproduced, with permission, appear on the Credits page in the endmatter of this textbook.

Microsoft® and Windows® are registered trademarks of the Microsoft Corporation in the U.S.A. and other countries. This book is not sponsored or endorsed by or affiliated with the Microsoft Corporation.

ISBN 10: 1-292-06124-3
ISBN 13: 978-1-29-206124-5 (Print)
ISBN 13: 978-1-29-206192-4 (PDF)

10 9 8 7 6 5 4 3 2 1
14 13 12 11 10

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Typeset in 9 JansonTextLTStd-Roman by Laserwords.

Printed and bound by Courier Kendallville.

The publisher's policy is to use paper manufactured from sustainable forests.



WELCOME to the Sixth Edition of *Fluency with Information Technology: Skills, Concepts, & Capabilities*. This book teaches the foundational ideas of computing. Today's students, having grown up in a world in which computers are ubiquitous, know how to use them, intuitively. But *use* is not understanding. Today's students belong to a world that requires them to be “computational thinkers,” able to conceptualize how computation can be used effectively. Few students will work as programmers, but most will decide how to apply computation. And many will imagine new ways for computers to serve humanity. To be effective at these tasks, they must understand the fundamental ideas of computing. This is their book.

What's New in This Edition?

More than any previous edition, this sixth edition illustrates a fundamental point about *Fluency*: Though the concepts are constant, their practical manifestations change at warp speed. Concept-wise, the sixth edition retains all of the fundamental ideas that underlie the Fluency vision. In practice, however, much of the text has been rewritten to accommodate how students encounter computation: advances in smartphones, HTML5, CSS3, new browser support for JavaScript programming, cloud technology, and on and on. These affect how students use and perceive the fundamentals. Topics like crowd sourcing, privacy, security, phishing, AI, netiquette, copyright, and so forth evolve, and so they must be explained in contemporary terms consistent with student experience. And “ambient knowledge” changes, too. For example, unlike the past today's students have at least heard a term like *algorithm*. But, awareness is not understanding, and so the sixth edition has been revised to explain the concept with a new, more intuitive approach. Other newly familiar terms have been similarly treated.

Readers familiar with earlier versions of *Fluency* should take a close look at this sixth edition. It maintains the core “fluency vision,” while positioning the presentation squarely in the second decade of the 21st century.

Briefly, the four parts of Fluency have been updated as follows:

- ▶ Part 1 has undergone a complete makeover. Chapter 1 “Defining IT,” Chapter 3 “Networking,” and Chapter 5 “Web” are (again) new. Chapter 4 “HTML” is redeveloped to teach HTML5 and CSS3. Chapter 2 “Human-Computer Interface” and Chapter 6 “Debugging” have been substantially revised.
- ▶ In Part 2, the “bits part” of Chapter 7 “Digital Information” has been redeveloped, Chapter 9 “Computer Organization” has been completely rewritten to be more intuitive, and Chapter 10 “Algorithms” is new and much simplified.

- ▶ In Part 3, Chapter 11 “Social Implications” is mostly new, and Chapter 12 “Privacy and Security” is completely revised to deal with recently revealed privacy threats (NSA) and ongoing security attacks. The “Spreadsheets” chapters (13 and 14) have been revised to increase compatibility with different implementations, including various Excel versions. Chapter 15 “Database Concepts” has a complete reformulation of the relational model. Chapter 16 the “iDiary Database” received a wholesale revision.
- ▶ Finally, in Part 4, the “JavaScript” chapters (17–21) have been redeveloped to use the Firefox Scratchpad sandbox for code development, a definite pedagogical advance. Chapter 22, the artificial intelligence chapter, has had its Watson discussion augmented by a new interview with David Ferrucci, the Watson project leader.

A number of “Try It” exercises and the end-of-chapter Review Questions have been heavily revised; new exercises have been added. Further resources are listed below.

***Fluency*—a Text for Computer Science Principles**

The vision of *Being Fluent with Information Technology* (see below) is to introduce fundamental computational concepts to college students; implementing that vision was the original purpose of this book. The Computer Science Principles vision is virtually identical, but for high school students. Think of a college Fluency course as the class that successful AP CS Principles students get credit for. It follows that *Fluency* is a suitable text for CS Principles.

Depending on specific choices of the instructor, the two curricula overlap about 85–90 percent in terms of concepts and ideas. Both cover algorithms, digital data and metadata, programming, Internet fundamentals, security and privacy, AI, and so on. The difference is more on emphasis than with content. The courses have been described as

“Fluency introduces computing ideas to enhance students’ ability to apply computation” and

“CS Principles introduces computing ideas for their scientific richness.”

Both are important (and not mutually exclusive) goals; I have ensured that *Fluency* fulfills both.

To illustrate, recall that the Computer Science Principles curriculum is guided by the Seven Big Ideas [www.csprinciples.org]. The content of *Fluency*’s chapters (sections or possibly the entire chapter) aligns with the big ideas as follows:

- ▶ Creativity—4, 5, 6, 10, 16, 18, 19, 20, 21
- ▶ Abstraction—1, 4, 10, 17, 18, 19, 20, 21
- ▶ Data—7, 8, 13, 14, 15, 16, 17, 22
- ▶ Algorithms—1, 2, 7, 8, 9, 10, 22
- ▶ Programming—4, 6, 9, 10, 16, 17, 18, 19, 20, 21, 22
- ▶ Internet—3, 4, 5, 8, 11, 12, 16, 19, 20, 21
- ▶ Global Impacts—1, 2, 3, 4, 5, 9, 11, 12, 22

The alignment is explained in finer grain detail in the Instructor’s Resources.

What Is Fluency with Information Technology?

This book is inspired by a report from the National Research Council (NRC), *Being Fluent with Information Technology*. In that study, commissioned by the National Science Foundation, the committee asserted that traditional computer literacy does not have the “staying power” students need to keep pace with the rapid changes in IT. The study concluded that the educational “bar needs to be raised” if students’ knowledge is to evolve and adapt to that change. The recommended alternative, dubbed *fluency with information technology*, or *FIT*, was a package of skills, concepts, and capabilities wrapped in a project-oriented learning approach that ensures that the content is fully integrated. The goal is to help people become effective IT users immediately and to prepare them for lifelong learning.

The Vision

This textbook largely implements the vision of the NRC committee in which they proposed a three-part content and recommended integrating that content by using projects.

Three-Part Content

To make students immediately effective and launch them on the path of lifelong learning, they need to learn three types of knowledge: Skills, Concepts, and Capabilities.

- ▶ **Skills** refers to proficiency with contemporary computer applications such as email, word processing, Web searches, and so forth. This knowledge, originally described as “literacy,” is now generally known to students on arrival at college; it need not be taught. Only “advanced skills”—essentially spreadsheets and database queries—remain in *Fluency*.
- ▶ **Concepts** refers to the fundamental knowledge underpinning IT, such as computer functionality, digital representation of information, assessment of information authenticity, and so forth. Concepts provide the principles on which students will build new understanding as IT evolves.
- ▶ **Capabilities** refers to higher-level thinking processes such as problem solving, reasoning, complexity management, and troubleshooting. Creativity is a key capability. Capabilities embody modes of thinking that are essential to exploiting IT, but they apply broadly. Reasoning, problem solving, and so forth are standard components of education; their significance in IT makes them topics of emphasis in the *Fluency* approach.

For each component, the NRC report lists 10 recommended items, which are covered as needed in this book.

Audience

This book is designed for freshmen “nontechies,” students who will not be majoring in science, engineering, or math. (“Techies” benefit too, but because “hot shots” can intimidate others, they should be discouraged from taking the class, or better, encouraged to join an accelerated track or honors section.) No mathematical skills are required beyond arithmetic. There are no prerequisites.

Chapter Dependencies

Fluency with Information Technology is organized so that it can be taught in a variety of ways. In addition to the preliminary material in Chapters 1 and 2, social issues in Chapters 11 and 12, and the wrap-up in Chapter 23, the overall structure of the book includes stand-alone chapters with few dependencies, as well as small chapter sequences devoted to a sustained topic. The sequences are as follows:

- ▶ Chapters 3, 4, and 5—networking, HTML, and information
- ▶ Chapters 7, 8, 9, and 10—data representations, computers, and algorithms
- ▶ Chapters 13, 14, 15, and 16—spreadsheets and database principles
- ▶ Chapters 17, 18, 19, 20, and 21—programming in JavaScript

One effective way to use this design is to present one of the chapter sequences as the basis for a project assignment. Then, while the students are working on the project—projects may span two or more weeks—material from stand-alone chapters is covered.

There are many sequences, but three stand out as especially effective to present the material:

- ▶ **Networking cycle.** The linear sequence of chapters is designed to begin with information and networking and progressively advance through computation and databases to JavaScript, where it returns to the networking theme. This is the basic Chapters 1 to 23 sequence, adjusted by local reordering to accommodate the timing of projects as needed.
- ▶ **Internet forward.** I teach *Fluency* in the 1–10, 17–21, 11–16, 22–23 order. This approach begins with information and HTML, progresses through to algorithms, then jumps to JavaScript to continue the Web page building theme, and finally wraps up with databases. The strategy is dictated largely by the logistics of teaching the class in a quarter (10 weeks) and is recommended for that situation.
- ▶ **Traditional.** In this approach, the material is taught to parallel the time sequence of its creation. So, information representation and computers come well before networking. In this case, the order is 1–2, (22), 7–16, 3–6, 17–23. Chapter 22, which contains more philosophical content like the Turing test, Kasparov/Deep Blue chess tournament, and Watson, might optionally be presented early for its foundational content.

Each of these strategies has a compelling pedagogical justification. The one chosen depends more on the instructor's taste and class logistics than on a need to present material in a specific order.

Pedagogical Features

- ▶ **Learning Objectives:** Each chapter opens with a list of the key concepts that readers should master after reading the chapter.

There are several boxed features that appear throughout the text to aid in students' understanding of the material:

- ▶  Practical hints and suggestions for everyday computer use

-  **and**  Interesting facts and statistics
-  Warnings and explanations of common mistakes
-  Short, in-chapter exercises with provided solutions

Throughout the text, notable material is distinguished by the following features:

-  A historical look at some of the major people and milestones in computing
- Glossary:** Important words and phrases are bold throughout the text, and a glossary of terms is included at the end of the book; glossary terms are bold in the book's index
- Answers:** Solutions are provided to selected questions for the multiple-choice and short-answer sections
- Appendix A:** XHTML reference including a chart of Web-safe colors
- Appendix B:** RSA public key cryptosystem
- Appendix C:** The XML database and the XSL template style information for iDiary in Chapter 16
- Appendix D:** JavaScript programming rules
- Appendix E:** Bean Counter Program: A complete JavaScript and HTML example
- Appendix F:** Memory Bank Page: A complete JavaScript and HTML example
- Appendix G:** Smooth Motion Application: A complete JavaScript and HTML example

Supplements

The Companion Web site for *Fluency with Information Technology* is located at:

www.pearsonglobaleditions.com/snyder

The following student supplements are available at the Web site:

- 23 online labs (1 for each chapter in the book).** The labs are designed to more fully explore (and test) students' understanding of concepts in the book and how those issues or concepts apply to their lives. For example, in Chapter 1 the lab explores the conflicting interests of searchable, interconnected information and privacy. It directs students to a variety of Web-based resources that can help them understand what kinds of information they (and their devices) are sharing, which can be surprisingly personal and sometimes can even uniquely identify the user. Understanding the scope and type of information sharing that is happening helps reinforce the overall theme of understanding

how things work “under the hood.” See the listing of labs at the end of the preface or see the complete set of labs at www.pearsonglobaleditions.com/snyder.

- ▶ **Computer Skills Workbook by Sharon Scollard.** This book covers the basics of Microsoft Office suite and includes 14 comprehensive labs on Excel, Word, PowerPoint, and other topics.
- ▶ **VideoNotes.** VideoNotes are short, tutorial videos that enhance concepts presented in the textbook.
- ▶ **HTML sources, database designs, and JavaScript programs used in the textbook examples**
- ▶ **JavaScript reference card**
- ▶ **Glossary flashcards**
- ▶ **A downloadable workbook (PDF) on Alice and the Alice development environment**

The following instructor supplements are available to qualified instructors only. Please contact your local Pearson Education representative for information on how to access them (you can find your rep at www.pearsonglobaleditions.com/snyder).

- ▶ **PowerPoint slides**
- ▶ **Instructor’s Solutions Manual**
- ▶ **Test Bank**
- ▶ **Test Generator**

Note to Students

Fluency is a somewhat unusual topic, making this a somewhat unusual book. I have one bit of advice to make it easier to learn this material.

- ▶ **Study *Fluency* steadily.** If this book is successful, it will change the way you think, making you better at problem solving, better at reasoning, better at debugging, and so forth. These Capabilities are useful in IT and elsewhere in life, so they make learning *Fluency* worthwhile. But changing how you think won’t happen by putting the book under your pillow. It’ll take some studying. To learn *Fluency* you must apply good study habits: read the book, do the end-of-chapter exercises (answers to selected questions appear at the back of the book), begin your assignments early, ask questions, and so on. I think it’s a good idea if you spend some time online studying *Fluency* (instead of surfing) every day, because it takes time for the ideas to sink in. Students with good study habits tend to do well in *Fluency* class, and because it improves their problem-solving abilities, and more, they become even better students! It takes some discipline but it pays.

Finally, reading this book is enhanced by having a computer handy so you can try the examples. The files used are available at www.pearsonglobaleditions.com/snyder. Good luck! Writing this book has truly been a pleasure. I hope reading it is equally enjoyable.

Acknowledgments

It is a pleasure to thank my collaborators in the creation of the *Fluency* concept, the NRC Committee on Computer Literacy—Al Aho, Marcia Linn, Arnie Packer, Allen Tucker, Jeff Ullman, and Andy van Dam. Special thanks go to Herb Lin of the NRC staff who assisted throughout the *Fluency* effort, tirelessly and in his usual great good humor. Two enthusiastic supporters of *Fluency*—Bill Wulf of the National Academy of Engineering and John Cherniavski of the National Science Foundation—have continually supported this effort in more ways than I am aware. It has been a pleasure to know and work with this team.

I am particularly grateful for the keen insights and valuable feedback from the reviewers of this book. For this edition, my thanks go to the following: Deborah Noonan (College of William & Mary), Jie Meichsner (Saint Cloud State University), Melissa Wiggins (Mississippi College), Lynn Thibodeau (University of Hartford), Paolina Centonze (Iona College), Timothy Highley (La Salle University), Thomas Bennet (Mississippi College).

Thank you to Susan Evans for contributing to the end-of-chapter Review Questions.

Thanks to Rebecca Greenberg, whose editing has improved the content and presentation significantly; as always it is a great pleasure to work with Gillian Hall. And, once again it is my pleasure to thank Matt Goldstein and the whole Pearson team, especially Jeff Holcomb and Marilyn Lloyd.

Finally, my wife, Julie, has been patient, encouraging, and, most important, a continual source of good humor throughout this effort. It is with my deepest appreciation that I thank her for everything.

—Larry Snyder, January 2014

Pearson Education wishes to thank Arup Bhattacharjee, Soumen Mukherjee, and Manasa S. of NMAM Institute of Technology, Nitte for reviewing the Global Edition.

For Julie

This page is intentionally left blank.



PART 1 Becoming Skilled at Computing

chapter 1 Defining Information Technology

Terms of Endearment 21

Computation's Greatest Hits 23

Terms of Endearment 32

Computers, Software, Algorithms 33

The Words for Ideas 37

Summary 40

Try It Solutions 40

Review Questions 40

chapter 2 Exploring the Human-Computer Interface

Face It, It's a Computer 43

A Few Useful Concepts 45

Perfect Reproduction 47

What We See and What We Think 52

Summary 56

Try It Solutions 56

Review Questions 57

chapter 3 The Basics of Networking

Making the Connection 61

Comparing Communication Types 63

The Medium of the Message 66

The World Wide Web 78

File Structure 81

Summary 85

Try It Solutions 85

Review Questions 86

chapter 4 A Hypertext Markup Language Primer

Marking Up with HTML 89

Marking Up with HTML 91

Lab Practice I 93

Structuring Documents 96

Lab Practice II 101

Get Into Style with CSS 103

Marking Links and Images 106

Referring to Files 109

Span, Lists, Tables, and Boxes 111

Cascading Style Sheets 117

Styling with Class 119

Hovering Above Links 122

HTML Wrap-Up 124

Summary 125

Try It Solutions 126

Review Questions 126

chapter 5 Locating Information on the WWW

The Search for Truth 131

Web Search Fundamentals 133

Advanced Searches 139

Web Searching 144
 Authoritative Information 149
 Truth or Fiction? 153

Summary 157
 Try It Solutions 157
 Review Questions 158

chapter 6 An Introduction to Debugging

To Err Is Human 161

Precision: The High Standards of
 Computing 163
 Debugging: What's the Problem? 163
 A Dialog About Debugging 165
 Debugging Recap 169
 Fixing HTML Bugs: A Case
 Study 170
 No Printer Output: A Classic
 Scenario 180

Ensuring the Reliability of
 Software 183
 Community Debugging 184
 Summary 185
 Try It Solutions 185
 Review Questions 186
 Interview with Vinton G. Cerf 189

PART 2 Algorithms and Digitizing Information

chapter 7 Representing Information Digitally

Bits and the "Why" of Bytes 195

Digitizing Discrete Information 197
 Information Representation 199
 Binary and Hex 203
 Digitizing Numbers in Binary 205
 Digitizing Text 208

UTF10 212
 The Metadata and the OED 213
 Summary 218
 Try It Solutions 218
 Review Questions 219

chapter 8 Representing Multimedia Digitally

Light, Sound, Magic 221

Digitizing Color 223
 Computing on Representations 230
 Digitizing Sound 236
 Digital Images and Video 239
 Optical Character Recognition 244

Multimedia Challenges 246
 Bits Are It 247
 Summary 248
 Try It Solutions 248
 Review Questions 249

chapter 9 Principles of Computer Operations

Following Instructions 253

There's an App for That 255
 Software Isn't So Hard 256
 Instruction Execution Engine 260
 The Program Counter:
 The PC's PC 266
 Instruction Execution 267
 Translation 272

Integrated Circuits 275
 How Semiconductor Technology
 Works 278
 Combining the Ideas 280
 Summary 282
 Try It Solutions 282
 Review Questions 283

chapter 10 Algorithmic Thinking

What's the Plan? 287

Algorithms 289
 Algorithms—A Basic Concept 295

A Closer Look 296
 Doing the Right Thing 300

Summary 303
 Try It Solutions 303

Review Questions 304
 Interview with Ray Kurzweil 306

PART 3 Data and Information

chapter 11 Social Implications of IT

Computers in Polite Society 313

The Power of the Crowd 315
 Out on Good Behavior 317
 Expect the Unexpected 321
 Creating Good Passwords 322
 Spam 327
 Scams 328

Protecting Intellectual Property 333
 Creative Commons 337
 Summary 339
 Try It Solutions 340
 Review Questions 340

chapter 12 Privacy and Digital Security

Shhh, It's a Secret 345

Privacy and Technology 347
 A Privacy Definition 348
 Enjoying the Benefits of Privacy 349
 Fair Information Practices 350
 Is There No Privacy? 351
 Tracking 354
 Cookies 358
 Digital Security 360

Prevention 363
 Encryption 367
 Redundancy Is Very, Very, Very
 Good 373
 Summary 376
 Try It Solutions 377
 Review Questions 377

chapter 13 The Basics of Spreadsheets

Fill-in-the-Blank Computing 381

Arranging Information 383
 Computing with Spreadsheets 387
 Daily Spreadsheets 396
 Importing Data 402

Summary 407
 Try It Solutions 408
 Review Questions 408

chapter 14 Advanced Spreadsheets for Planning

"What If" Thinking Helps 413

Designing a Spreadsheet 415
 Conditional Formatting 418
 Conditional Formulas 421
 Naming: Symbolic Reference 424
 "What If" Analysis 427

Analyzing Data Using Filtering 433
 Summary 437
 Try It Solutions 438
 Review Questions 438

chapter 15 Introduction to Database Concepts

A Table with a View 441

Differences Between Tables and
 Databases 443
 XML: A Language for Metadata
 Tags 444
 Tables and Entities 450
 The Science of Tables 453

SQL: The Language of Databases 464
 Structure of a Database 467
 Summary 469
 Try It Solutions 469
 Review Questions 470

chapter **16** **A Case Study in Database Organization**

- The iDiary Database 473
- Thinking About a Personal Database 475
- A Preliminary Exercise 476
- The iDiary Database 486
- Using the iDiary Daily 496
- Summary 498
- Try It Solutions 498
- Review Questions 499
- Interview with Alan Kay 501

PART 4 **Problem Solving**chapter **17** **Fundamental Concepts Expressed in JavaScript**

- Get with the Program 507
- Overview: Programming Concepts 509
- Names, Values, and Variables 510
- Three Basic Data Types of JavaScript 514
- The Assignment Statement 517
- Lab Practice 520
- An Expression and Its Syntax 522
- A Conditional Statement 526
- The Espresso Program 531
- Summary 533
- Try It Solutions 534
- Review Questions 535

chapter **18** **A JavaScript Program**

- The Bean Counter 539
- Preliminaries 541
- Background for the UI 542
- Creating the Graphical User Interface 545
- Event-Based Programming 548
- Critiquing the Bean Counter 552
- Bean Counter Recap 555
- Summary 556
- Try It Solutions 556
- Review Questions 557

chapter **19** **Programming Functions**

- Thinking Big 561
- Anatomy of a Function 563
- Forms and Functions 566
- Writing Functions, Using Functions 568
- Customizing Pages 574
- Making a Web-Based Phone App 577
- Social Functions 582
- Summary 588
- Try It Solutions 589
- Review Questions 590

chapter **20** **Iteration Principles**

- Once Is Not Enough 593
- Iteration: *Play It Again, Sam* 595
- JavaScript Rules for for Loops 597
- Experiments with Flipping Coins 600
- Indexing 605
- Arrays 606
- It's Magic 608
- The Busy Animation 610
- Not So Busy Animation 615
- Summary 617
- Try It Solutions 618
- Review Questions 620

chapter **21** **A Case Study in Algorithmic Problem Solving**

- The Smooth Motion Application 623
- The Smooth Motion Application 625
- Planning Smooth Motion 626

Build the Basic Web Page UI 629
 Animate the Grid 630
 The Best Laid Plans . . . 634
 Build Controls 635
 Sense the Keys 636
 Staircase Detection 639

Assemble Overall Design 640
 Primp the Design 642
 Summary 643
 Try It Solutions 644
 Review Questions 645

chapter **22** Limits to Computation

Computers Can Do Almost {□ Everything, □ Nothing} 647

Can Computers Think? 649
 Acting Intelligently? 651
 Watson 656
 Acting Creatively? 660
 The Universality Principle 663

More Work, Slower Speed 666
 Are “Best” Algorithms All Fast? 667
 Summary 669
 Try It Solutions 670
 Review Questions 670

chapter **23** A Fluency Summary

Click to Close 673

Two Big Computing Ideas 675
 Fluency: Less Is More 676
 Lifelong IT Learning 678
 Shifting for Yourself 680

Try It Solutions 680
 Review Questions 680
 Interview with David Ferrucci 682

appendix **A** HTML5 Reference 687

appendix **B** RSA Public Key Cryptosystem 693

appendix **C** iDiary: Tags and Templates 699

appendix **D** JavaScript Programming Rules 705

appendix **E** The Bean Counter Program 713

appendix **F** myApps Page 717

appendix **G** Smooth Motion Program 725

Glossary 729

Answers to Selected Questions 739

Index 749

Credits 769

Location of VideoNotes in the Text 	
Chapter 1	The Right Way to Say IT, p. 33
Chapter 2	Placeholders Gone Wrong, p. 50
Chapter 3	Transportation Networks, p. 71 Internet Traffic Signals, p. 79
Chapter 5	Operators and Search, p. 140 Understanding Search Engines, p. 144 Books Versus Web Sites, p. 149
Chapter 6	Debugging Your Math Homework, p. 164 The Case of the Missing Bracket, p. 177
Chapter 7	The PandA Problem, p. 200 ASCII Hidden Messages, p. 209 Binary to ASCII Decoding, p. 210
Chapter 9	Breaking IT Down, p. 267
Chapter 10	Creative Communication, p. 292 One Problem, Many Algorithms, p. 300
Chapter 11	How Many Passwords? p. 324
Chapter 12	The Cookies That Knew Too Much, p. 358
Chapter 13	Repaying a Loan, p. 402 How Your Money <i>Could</i> Grow, p. 402
Chapter 14	Quiz Bank, p. 436 Your Class Schedule, Spreadsheet Style, p. 436
Chapter 15	Your Class Schedule, XML Style, p. 450
Chapter 17	Stepping Through the Espresso Program, p. 531
Chapter 18	What a Click Event Does, p. 549
Chapter 19	A Function for Calculating Wages, p. 568
Chapter 20	Looping from 1 to 100, p. 608
Chapter 22	Game Trees, p. 654
Chapter 23	Exit Conditions, p. 676

Online Labs

This series of labs is designed to help students understand different perspectives on technology, to get hands-on experience with tools, and to think about what it means to live in a world where more and more of what they do on a daily basis involves using technology.

Each lab has an introduction, a “To Consider” section, a set of exercises, and a “Moving On” summary. Once students become familiar with one lab, they’ll know what to expect in the others.

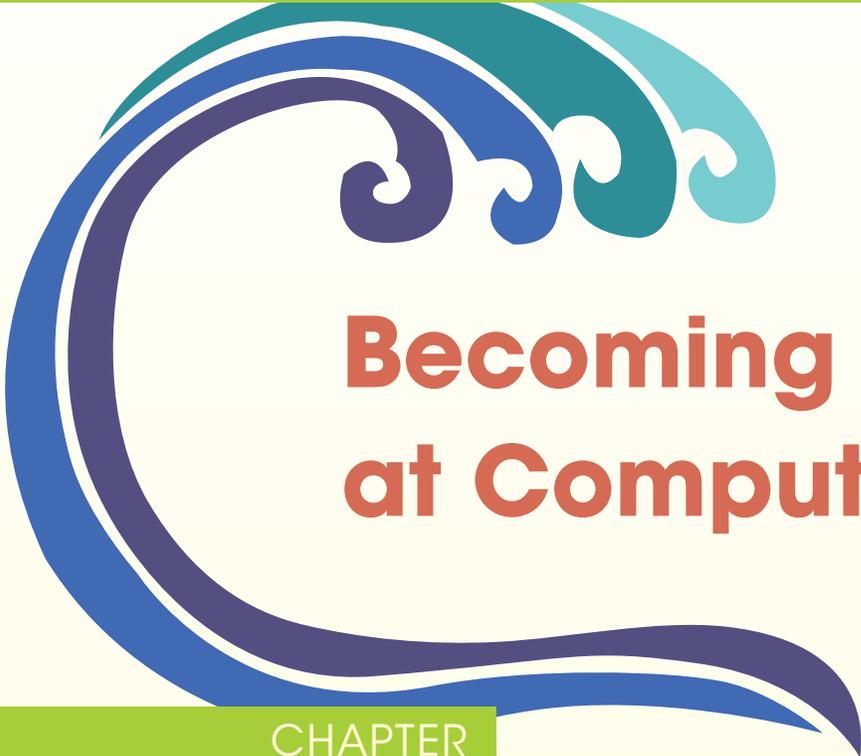
Labs can be accessed at www.pearsonglobaleditions.com/snyder

Lab 00	Introduction and Resources
Lab 01	Information on the Web
Lab 02	Human Computer Interfaces
Lab 03	How the Internet Is Connected
Lab 04	HTML Introduction
Lab 05	Search
Lab 06	Debugging
Lab 07	Digital Encoding
Lab 08	Digital Media
Lab 09	The Fetch/Execute Cycle
Lab 10	Algorithmic Thinking
Lab 11	Collaboration over the Web
Lab 12	Privacy
Lab 13	Introduction to Spreadsheets
Lab 14	Data and Information Visualizations
Lab 15	Introduction to Databases
Lab 16	Adding an RSS Feed to the iDiary
Lab 17	Introduction to JavaScript
Lab 18	Extending the Bean Counter Program

Lab 19	Creating JavaScript Functions
Lab 20	JavaScript Loops
Lab 21	More JavaScript (Interactivity)
Lab 22	AI and Androids
Lab 23	Keeping Current with IT

The Master said: "To learn something and then put it into practice at the right time. Is this not a joy?"

—THE ANALECTS OF CONFUCIUS



Becoming Skilled at Computing

CHAPTER

- 1 Defining Information Technology
- 2 Exploring the Human-Computer Interface
- 3 The Basics of Networking
- 4 A Hypertext Markup Language Primer
- 5 Locating Information on the WWW
- 6 An Introduction to Debugging

PART 1

INTRODUCTION

OUR STUDY of computing assumes that you have used computers your whole life. Being an accomplished computer user means that you have a general idea of how to make computing systems work for you, so there is no need to teach you to be a user. Instead, we will explain “what’s happening” under the glitzy graphics and “how it works” behind the magic. Most of us wonder from time to time what makes the technology click, and that is reason enough to study this material. But what you learn in this class will almost certainly make you a better and more knowledgeable user. So whether it’s curiosity or knowhow that drives your interest, learning computing more fully will pay off.

Part 1 focuses on setting a firm foundation for the future. The Internet is everywhere and consumes our lives. We need to know more about it, how it works, and how to use it effectively. We begin by setting the context for further study, and explaining why you do what you do intuitively. We study how networking works, try our hand at HTML, tune-up our searching skills, and practice debugging. These topics will get us off to a fast start.