STARTING OUT WITH JAVA™

From Control Structures through Objects

STARTING OUT WITH JAVA™

From Control Structures through Objects

SEVENTH EDITION

Tony Gaddis
Haywood Community College

Pearson
330 Hudson Street, NY NY 10013

Senior Vice President Courseware Portfolio Management: Marcia J. Horton
Director, Portfolio Management: Engineering, Computer Science & Global Editions: Julian Partridge
Portfolio Manager: Matt Goldstein
Portfolio Management Assistant: Meghan Jacoby
Managing Content Producer: Scott Disanno
Content Producer: Amanda Brands
Rights and Permissions Manager: Ben Ferrini
Manufacturing Buyer, Higher Ed, Lake Side Communications, Inc. (LSC): Maura Zaldivar-Garcia
Inventory Manager: Ann Lam
Product Marketing Manager: Yvonne Vannatta
Field Marketing Manager: Demetrius Hall
Marketing Assistant: Jon Bryant
Cover Designer: Joyce Wells
Cover Photo: Shutterstock/Tim UR
Printer/Binder: LSC Communications, Inc.
Full-Service Project Management: Sasibalan Chidambaram, SPi Global

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on appropriate page within text.

Copyright © 2019, 2016, 2013 by Pearson Education, Inc., Hoboken, New Jersey 07030. All rights reserved. Manufactured in the United States of America. This publication is protected by copyright and permissions should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms and the appropriate contacts within the Pearson Education Global Rights & Permissions department, please visit http://www.pearsoned.com/permissions/.

Many of the designations by manufacturers and seller to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Unless otherwise indicated herein, any third-party trademarks that may appear in this work are the property of their respective owners and any references to third-party trademarks, logos or other trade dress are for demonstrative or descriptive purposes only. Such references are not intended to imply any sponsorship, endorsement, authorization, or promotion of Pearson's products by the owners of such marks, or any relationship between the owner and Pearson Education, Inc. or its affiliates, authors, licensees or distributors.

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability. Whether express, implied or statutory, fitness for a particular purpose, title and non infringement. In no event shall microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract. Negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time partial screen shots may be viewed in full within the software version specified.

Microsoft® Windows®, and Microsoft Office® 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.

Library of Congress Cataloging-in-Publication Data

Names: Gaddis, Tony, author.
Title: Starting out with Java. From control structures through objects / Tony Gaddis, Haywood Community College.
Description: Seventh edition. | NY, NY : Pearson Education, Inc., [2019] | Includes index.
Identifiers: LCCN 2017060354| ISBN 9780134802213 | ISBN 0134802217
Subjects: LCSH: Java (Computer program language) | Data structures (Computer science) | Object-oriented programming (Computer science)
Classification: LCC QA76.73.J38 G333 2019 | DDC 005.13/3--dc23 LC record available at https://lccn.loc.gov/2017060354

1 18

Pearson

ISBN 10: 0-13-480221-7
ISBN 13: 978-0-13-480221-3

Contents in Brief

Contents

Chapter 5 **Methods 269**

Chapter 6 A First Look at Classes 317

Chapter 7 Arrays and the `ArrayList` Class 403

Chapter 8 **A Second Look at Classes and Objects 493**

Chapter 9 **Text Processing and More about Wrapper Classes 557**

Chapter 10 **Inheritance 611**

Chapter 11 **Exceptions and Advanced File I/O 701**

Chapter 12 **JavaFX: GUI Programming and Basic Controls 759**

Chapter 14 **JavaFX: Graphics, Effects, and Media 909**

Chapter 16 **Databases 1027**

Index 1109

The following appendices, online chapters, and online case studies are available on the book's online resource page at www.pearson.com/cs-resources.

Online Appendices:

Online Chapters:

Online Case Studies:

LOCATION OF VIDEONOTES IN THE TEXT

Chapter 1	Compiling and Running a Java Program, p. 14 Using an IDE, p. 14 Your First Java Program, p. 25
Chapter 2	Displaying Console Output, p. 33 Declaring Variables, p. 39 Simple Math Expressions, p. 55 The Miles-per-Gallon Problem, p. 107
Chapter 3	The `if` Statement, p. 111 The `if-else` Statement, p. 120 The `if-else-if` Statement, p. 129 The Time Calculator Problem, p. 182
Chapter 4	The `while` Loop, p. 193 The Pennies for Pay Problem, p. 263
Chapter 5	Passing Arguments to a Method, p. 279 Returning a Value from a Method, p. 293 The Retail Price Calculator Problem, p. 311
Chapter 6	Writing Classes and Creating Objects, p. 325 Initializing an Object with a Constructor, p. 346 The Personal Information Class Problem, p. 395
Chapter 7	Accessing Array Elements in a Loop, p. 407 Passing an Array to a Method, p. 422 The Charge Account Validation Problem, p. 487
Chapter 8	Returning Objects from Methods, p. 503 Aggregation, p. 515 The `BankAccount`, Class Copy Constructor Problem, p. 552
Chapter 9	The Sentence Capitalizer Problem, p. 605
Chapter 10	Inheritance, p. 611 Polymorphism, p. 655 The `Employee` and `Productionworker` Classes Problem, p. 696
Chapter 11	Handling Exceptions, p. 701 The Exception Project Problem, p. 757

(*continued on the next page*)

LOCATION OF VIDEONOTES IN THE TEXT *(continued)*

Chapter 12	Introduction to JavaFX, p. 762 Creating Scenes, p. 765 Displaying Images, p. 772 The HBox Layout Container, p. 777 The VBox Layout Container, p. 782 The GridPane Layout Container, p. 784 Button Controls and Events, p. 792 The TextField Control, p. 799 Using Anonymous Inner Classes as Event Handlers, p. 803 Using Lambda Expressions as Event Handlers, p. 806 The Latin Translator Problem, p. 818
Chapter 13	JavaFX and CSS, p. 823 RadioButton Controls, p. 838 CheckBox Controls, p. 848 ListView Controls, p. 853 ComboBox Controls, p. 874 Slider Controls, p. 880, The Dorm and Meal Plan Calculator Problem, p. 906
Chapter 14	Drawing Shapes with JavaFX, p. 909 JavaFX Animation, p. 940 JavaFX Effects, p. 958 Playing Sound Files with JavaFX, p. 970 Playing Videos with JavaFX, p. 974 Handling Key Events in JavaFX, p. 979 Handling Mouse Events in JavaFX, p. 986 The Coin Toss Problem, p. 996
Chapter 15	Reducing a Problem with Recursion, p. 1003 The Recursive Power Problem, p. 1026
Chapter 16	The Customer Inserter Problem, p. 1107

Preface

Welcome to *Starting Out with Java: From Control Structures through Objects*, Seventh Edition. This book is intended for a one-semester or a two-quarter CS1 course. Although it is written for students with no prior programming background, even experienced students will benefit from its depth of detail.

Control Structures First, Then Objects

This text first introduces the student to the fundamentals of data types, input and output, control structures, methods, and objects created from standard library classes.

Next, the student learns to use arrays of primitive types and reference types. After this, the student progresses through more advanced topics, such as inheritance, polymorphism, the creation and management of packages, GUI applications, recursion, and database programming. From early in the book, applications are documented with javadoc comments. As the student progresses through the text, new javadoc tags are covered and demonstrated.

As with all the books in the *Starting Out With* … series, the hallmark of this text is its clear, friendly, and easy-to-understand writing. In addition, it is rich in example programs that are concise and practical.

Changes in the Seventh Edition

This book's pedagogy, organization, and clear writing style remain the same as in the previous edition. The most significant change in this edition is the switch from Swing to JavaFX in the chapters that focus on GUI development. Although Swing is not officially deprecated, Oracle has announced that JavaFX has replaced Swing as the standard GUI library for Java.[1]

In this edition, we have added the following new chapters:

- **Chapter 12 JavaFX: GUI Programming and Basic Controls** This chapter presents the basics of developing graphical user interface (GUI) applications with JavaFX. Fundamental controls, layout containers, and the basic concepts of event-driven programming are covered.
- **Chapter 13 JavaFX: Advanced Controls** This chapter discusses CSS styling and advanced user interface controls.

[1] http://www.oracle.com/technetwork/java/javafx/overview/faq-1446554.html#6

- **Chapter 14 JavaFX: Graphics, Effects, and Media** This chapter discusses 2D shapes, animation, visual effects, playing audio and video, and responding to mouse and keyboard events.

The Swing and Applet material that appeared in the previous edition is still available on the book's companion Web site, as the following online chapters:

- The previous Chapter 12 *A First Look At GUI Applications* is now available online as Chapter 17.
- The previous Chapter 13 *Advanced GUI Applications* is now available online as Chapter 18.
- The previous Chapter 14 *Applets and More* is now available online as Chapter 19.

Note: Chapter 15 from the previous edition has also been moved to the book's companion Web site as *Chapter 20 Creating JavaFX Applications with Scene Builder*. Although Oracle no longer officially supports Scene Builder, it is still available as an open source tool at http:// http://gluonhq.com/labs/scene-builder/

In addition to the new JavaFX chapters, the Database chapter, which is now Chapter 15, has been updated to use JavaFX instead of Swing for its GUI applications. We have also added several new, motivational programming problems throughout the book.

Organization of the Text

The text teaches Java step-by-step. Each chapter covers a major set of topics and builds knowledge as students progress through the book. Although the chapters can be easily taught in their existing sequence, there is some flexibility. Figure P-1 shows chapter dependencies. Each box represents a chapter or a group of chapters. An arrow points from a chapter to the chapter that must be previously covered.

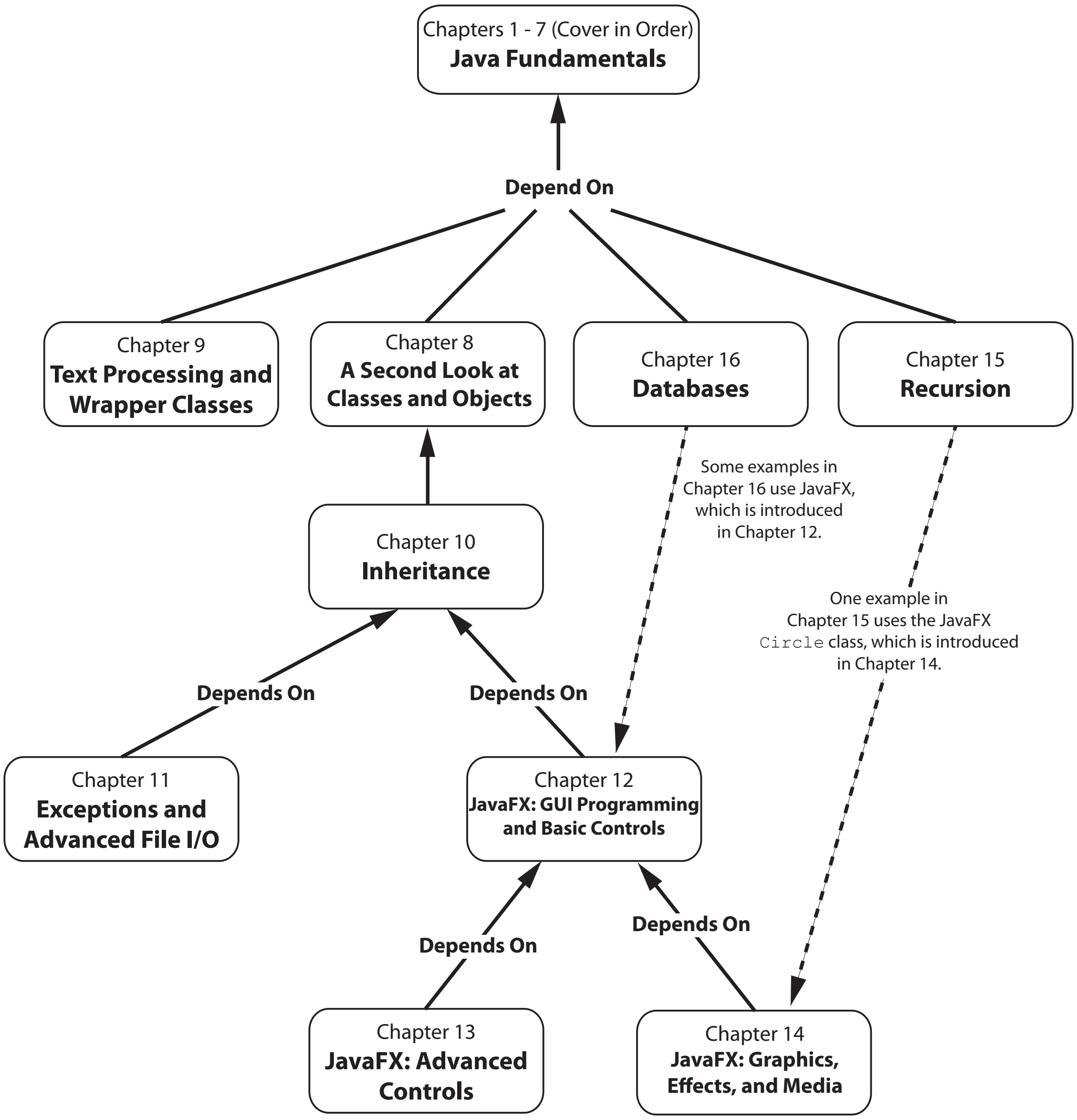
Brief Overview of Each Chapter

Chapter 1: Introduction to Computers and Java. This chapter provides an introduction to the field of computer science and covers the fundamentals of hardware, software, and programming languages. The elements of a program, such as key words, variables, operators, and punctuation, are discussed by examining a simple program. An overview of entering source code, compiling, and executing a program is presented. A brief history of Java is also given.

Chapter 2: Java Fundamentals. This chapter gets students started in Java by introducing data types, identifiers, variable declarations, constants, comments, program output, and simple arithmetic operations. The conventions of programming style are also introduced. Students learn to read console input with the `Scanner` class and with dialog boxes using `JOptionPane`.

Chapter 3: Decision Structures. In this chapter students explore relational operators and relational expressions and are shown how to control the flow of a program with the `if`, `if-else`, and `if-else-if` statements. Nested if statements, logical operators, the conditional operator, and the switch statement are also covered. The chapter discusses how to compare `String` objects with the `equals`, `compareTo`, `equalsIgnoreCase`, and `compareToIgnoreCase`

Figure P-1 Chapter dependencies


Chapters 1 - 7 (Cover in Order)
Java Fundamentals
Depend On
Chapter 9
Text Processing and Wrapper Classes
Chapter 8
A Second Look at Classes and Objects
Chapter 16
Databases
Chapter 15
Recursion
Some examples in Chapter 16 use JavaFX, which is introduced in Chapter 12.
Chapter 10
Inheritance
One example in Chapter 15 uses the JavaFX Circle class, which is introduced in Chapter 14.
Depends On
Depends On
Chapter 11
Exceptions and Advanced File I/O
Chapter 12
JavaFX: GUI Programming and Basic Controls
Depends On
Depends On
Chapter 13
JavaFX: Advanced Controls
Chapter 14
JavaFX: Graphics, Effects, and Media

methods. Formatting numeric output with the `System.out.printf` method and the `String.format` method is discussed.

Chapter 4: Loops and Files. This chapter covers Java's repetition control structures. The `while loop`, `do-while loop`, and `for loop` are taught, along with common uses for these devices. Counters, accumulators, running totals, sentinels, and other application-related topics are discussed. Simple file operations for reading and writing text files are included.

Chapter 5: Methods. In this chapter students learn how to write void methods, value-returning methods, and methods that do and do not accept arguments. The concept of functional decomposition is discussed.

Chapter 6: A First Look at Classes. This chapter introduces students to designing classes for the purpose of instantiating objects. Students learn about class fields and methods, and UML diagrams are introduced as a design tool. Then constructors and overloading are discussed. A `BankAccount` class is presented as a case study, and a section on object-oriented design is included. This section leads the students through the process of identifying classes and their responsibilities within a problem domain. There is also a section that briefly explains packages and the `import` statement.

Chapter 7: Arrays and the `ArrayList` Class. In this chapter students learn to create and work with single and multi-dimensional arrays. Numerous array-processing techniques are demonstrated, such as summing the elements in an array, finding the highest and lowest values, and sequentially searching an array. Other topics, including ragged arrays and variable-length arguments (varargs), are also discussed. The `ArrayList` class is introduced, and Java's generic types are briefly discussed and demonstrated.

Chapter 8: A Second Look at Classes and Objects. This chapter shows students how to write classes with added capabilities. Static methods and fields, interaction between objects, passing objects as arguments, and returning objects from methods are discussed. Aggregation and the "has a" relationship is covered, as well as enumerated types. A section on object-oriented design shows how to use CRC cards to determine the collaborations among classes.

Chapter 9: Text Processing and More about Wrapper Classes. This chapter discusses the numeric and `Character` wrapper classes. Methods for converting numbers to strings, testing the case of characters, and converting the case of characters are covered. Autoboxing and unboxing are also discussed. More `String` class methods are covered, including using the `split` method to tokenize strings. The chapter also covers the `StringBuilder` class.

Chapter 10: Inheritance. The study of classes continues in this chapter with the subjects of inheritance and polymorphism. The topics covered include superclasses, subclasses, how constructors work in inheritance, method overriding, polymorphism and dynamic binding, protected and package access, class hierarchies, abstract classes, abstract methods, anonymous inner classes, interfaces, and lambda expressions.

Chapter 11: Exceptions and Advanced File I/O. In this chapter students learn to develop enhanced error trapping techniques using exceptions. Handling exceptions is covered, as well as developing and throwing custom exceptions. The chapter discusses advanced techniques for working with sequential access, random access, text, and binary files.

Chapter 12: JavaFX: GUI Programming and Basic Controls. This chapter presents the basics of developing graphical user interface (GUI) applications with JavaFX. Fundamental controls, layout containers, and the basic concepts of event-driven programming are covered.

Chapter 13: JavaFX: Advanced Controls. This chapter discusses CSS styling and advanced user interface controls, such as RadioButtons, CheckBoxes, ListViews, ComboBoxes, Sliders, and TextAreas. Menu systems and FileChoosers are also covered.

Chapter 14: JavaFX: Graphics, Effects, and Media. This chapter discusses 2D shapes, animation, visual effects, playing audio and video, and responding to mouse and keyboard events.

Chapter 15: Recursion. This chapter presents recursion as a problem-solving technique. Numerous examples of recursive methods are demonstrated.

Chapter 16: Databases. This chapter introduces the student to database programming. The basic concepts of database management systems and SQL are first introduced. Then the student learns to use JDBC to write database applications in Java. Relational data is covered, and numerous example programs are presented throughout the chapter.

Features of the Text

Concept Statements. Each major section of the text starts with a concept statement that concisely summarizes the focus of the section.

Example Programs. The text has an abundant number of complete and partial example programs, each designed to highlight the current topic. In most cases the programs are practical, real-world examples.

Program Output. Each example program is followed by a sample of its output, which shows students how the program functions.

Checkpoints. Checkpoints, highlighted by the checkmark icon, appear at intervals throughout each chapter. They are designed to check students' knowledge soon after learning a new topic. Answers for all Checkpoint questions are provided in Appendix K, which can be downloaded from the book's resource page at www.pearson.com/cs-resources.

NOTE: Notes appear at several places throughout the text. They are short explanations of interesting or often misunderstood points relevant to the topic at hand.

TIP: Tips advise the student on the best techniques for approaching different programming problems and appear regularly throughout the text.

WARNING! Warnings caution students about certain Java features, programming techniques, or practices that can lead to malfunctioning programs or lost data.

In the Spotlight. Many of the chapters provide an *In the Spotlight* section that presents a programming problem, along with detailed, step-by-step analysis showing the student how to solve it.

VideoNotes. A series of videos, developed specifically for this book, are available at www.pearson.com/cs-resources. Icons appear throughout the text alerting the student to videos about specific topics.

Case Studies. Case studies that simulate real-world business applications are introduced throughout the text and are provided on the book's resource page at www.pearson.com/cs-resources.

Common Errors to Avoid. Each chapter provides a list of common errors and explanations of how to avoid them.

Review Questions and Exercises. Each chapter presents a thorough and diverse set of review questions and exercises. They include Multiple Choice and True/False, Find the Error, Algorithm Workbench, and Short Answer.

Programming Challenges. Each chapter offers a pool of programming challenges designed to solidify students' knowledge of topics at hand. In most cases the assignments present real-world problems to be solved.

Supplements

Student Online Resources

Many student resources are available for this book from the publisher. The following items are available on the Gaddis Series resource page at www.pearson.com/cs-resources:

- The source code for each example program in the book
- Access to the book's companion VideoNotes
- Appendixes A–M (listed in the Contents)
- A collection of seven valuable Case Studies (listed in the Contents)
- Online Chapters 17–20 (listed in the Contents)
- Links to download the Java™ Development Kit
- Links to download numerous programming environments including jGRASP™, Eclipse™, TextPad™, NetBeans™, JCreator, and DrJava

Online Practice and Assessment with MyLab Programming

MyLab Programming helps students fully grasp the logic, semantics, and syntax of programming. Through practice exercises and immediate, personalized feedback, MyLab Programming improves the programming competence of beginning students, who often struggle with the basic concepts and paradigms of popular high-level programming languages. A self-study and homework tool, the MyLab Programming course consists of hundreds of small practice exercises organized around the structure of this textbook. For students, the system automatically detects errors in the logic and syntax of their code submissions and offers targeted hints that enable students to figure out what went wrong—and why. For instructors, a comprehensive gradebook tracks correct and incorrect answers and stores the code inputted by students for review.

MyLab Programming is offered to users of this book in partnership with Turing's Craft, the makers of the CodeLab interactive programming exercise system. For a full demonstration, to see feedback from instructors and students, or to get started using MyLab Programming in your course, visit www.pearson.com/mylab/programming.

Instructor Resources

The following supplements are available to qualified instructors:

- Answers to all of the Review Questions
- Solutions for the Programming Challenges
- PowerPoint Presentation slides for each chapter
- Computerized Test Banks
- Source Code
- Lab Manual
- Student Files for the Lab Manual
- Solutions to the Lab Manual

Visit the Pearson Instructor Resource Center (www.pearson.com) or contact your local Pearson representative for information on how to access these resources.

Acknowledgments

There have been many helping hands in the development and publication of this book. We would like to thank the following faculty reviewers for their helpful suggestions and expertise:

Reviewers For This Edition

John Bono
George Mason University

Irene Bruno
George Mason University

Jackie Horton
University of Vermont

Mohammad T. Islam
Southern Connecticut State University

David Krebs
University of Pittsburgh

Reviewers of Previous Editions

Ahmad Abuhejleh
University of Wisconsin, River Falls

Colin Archibald
Valencia Community College

Ijaz Awani
Savannah State University

Bill Bane
Tarleton State University

N. Dwight Barnette
Virginia Tech

Asoke Bhattacharyya
Saint Xavier University, Chicago

Marvin Bishop
Manhattan College

Heather Booth
University of Tennessee, Knoxville

David Boyd
Valdosta University

Julius Brandstatter
Golden Gate University

Kim Cannon
Greenville Tech

Jesse Cecil
College of the Siskiyous

James Chegwidden
Tarrant County College

Kay Chen
Bucks County Community College

Brad Chilton
Tarleton State University

Diane Christie
University of Wisconsin, Stout

Cara Cocking
Marquette University

Jose Cordova
University of Louisiana, Monroe

Walter C. Daugherity
Texas A & M University

Michael Doherty
University of the Pacific

Jeanne M. Douglas
University of Vermont

Sander Eller
California Polytechnic University,
Pomona

Brooke Estabrook-Fishinghawk
Mesa Community College

Mike Fry
Lebanon Valley College

David Goldschmidt
College of St. Rose

Georgia R. Grant
College of San Mateo

Carl Stephen Guynes
University of North Texas

Nancy Harris
James Madison University

Chris Haynes
Indiana University

Ric Heishman
Northern Virginia Community College

Deedee Herrera
Dodge City Community College

Mary Hovik
Lehigh Carbon Community College

Brian Howard
DePauw University

Alan Jackson
Oakland Community College (MI)

Norm Jacobson
University of California, Irvine

Zhen Jiang
West Chester University

Stephen Judd
University of Pennsylvania

Neven Jurkovic
Palo Alto College

Dennis Lang
Kansas State University

Jiang Li
Austin Peay State University

Harry Lichtbach
Evergreen Valley College

Michael A. Long
California State University, Chico

Cheng Luo
Coppin State University

Tim Margush
University of Akron

Blayne E. Mayfield
Oklahoma State University

Scott McLeod
Riverside Community College

Dean Mellas
Cerritos College

Georges Merx
San Diego Mesa College

Martin Meyers
California State University,
Sacramento

Pati Milligan
Baylor University

Laurie Murphy
Pacific Lutheran University

Steve Newberry
Tarleton State University

Lynne O'Hanlon
Los Angeles Pierce College

Merrill Parker
Chattanooga State Technical Community College

Bryson R. Payne
North Georgia College and State University

Rodney Pearson
Mississippi State University

Peter John Polito
Springfield College

Charles Robert Putnam
California State University, Northridge

Y. B. Reddy
Grambling State University

Elizabeth Riley
Macon State College

Felix Rodriguez
Naugatuck Valley Community College

Diane Rudolph
John A Logan College

Carolyn Schauble
Colorado State University

Bonnie Smith
Fresno City College

Daniel Spiegel
Kutztown University

Caroline St. Clair
North Central College

Karen Stanton
Los Medanos College

Timothy Urness
Drake University

Peter van der Goes
Rose State College

Tuan A Vo
Mt. San Antonio College

Xiaoying Wang
University of Mississippi

Yu Wu
University of North Texas

Zijiang Yang
Western Michigan University

I also want to thank everyone at Pearson for making the *Starting Out With* … series so successful. I have worked so closely with the team at Pearson that I consider them among my closest friends. I am extremely fortunate to have Matt Goldstein as my editor, and Meghan Jacoby as Editorial Assistant. They have guided me through the process of revising this book, as well as many others. I am also fortunate to have Demetrius Hall as my marketing manager. His hard work is truly inspiring, and he does a great job of getting this book out to the academic community. The production team, led by Amanda Brands, worked tirelessly to make this book a reality. Thanks to you all!

About the Author

Tony Gaddis is the principal author of the *Starting Out With* … series of textbooks. He has nearly two decades of experience teaching computer science courses, primarily at Haywood Community College. Tony is a highly acclaimed instructor who was previously selected as the North Carolina Community College "Teacher of the Year" and has received the Teaching Excellence award from the National Institute for Staff and Organizational Development. The *Starting Out With* … series includes introductory textbooks covering programming logic and design, C++, Java™, Microsoft® Visual Basic®, Microsoft® Visual C#, Python, Alice, and App Inventor, all published by Pearson.

MyLab Programming

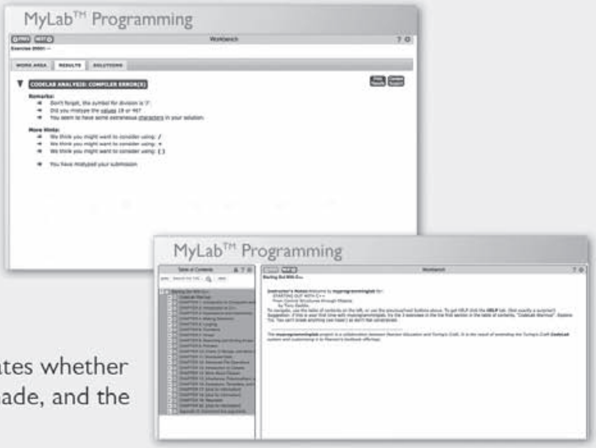
Through the power of practice and immediate personalized feedback, MyLab™ Programming helps students master programming fundamentals and build computational thinking skills.

PROGRAMMING PRACTICE

With MyLab Programming, your students will gain first-hand programming experience in an interactive online environment.

IMMEDIATE, PERSONALIZED FEEDBACK

MyLab Programming automatically detects errors in the logic and syntax of their code submission and offers targeted hints that enables students to figure out what went wrong and why.



GRADUATED COMPLEXITY

MyLab Programming breaks down programming concepts into short, understandable sequences of exercises. Within each sequence the level and sophistication of the exercises increase gradually but steadily.

DYNAMIC ROSTER

Students' submissions are stored in a roster that indicates whether the submission is correct, how many attempts were made, and the actual code submissions from each attempt.

PEARSON eTEXT

The Pearson eText gives students access to their textbook anytime, anywhere.

STEP-BY-STEP VIDEONOTE TUTORIALS

These step-by-step video tutorials enhance the programming concepts presented in select Pearson textbooks.

For more information and titles available with **MyLab Programming,** please visit **www.pearson.com/mylab/programming**

Copyright © 2018 Pearson Education, Inc. or its affiliate(s). All rights reserved. HELO88173 • 11/15

ALWAYS LEARNING

Pearson