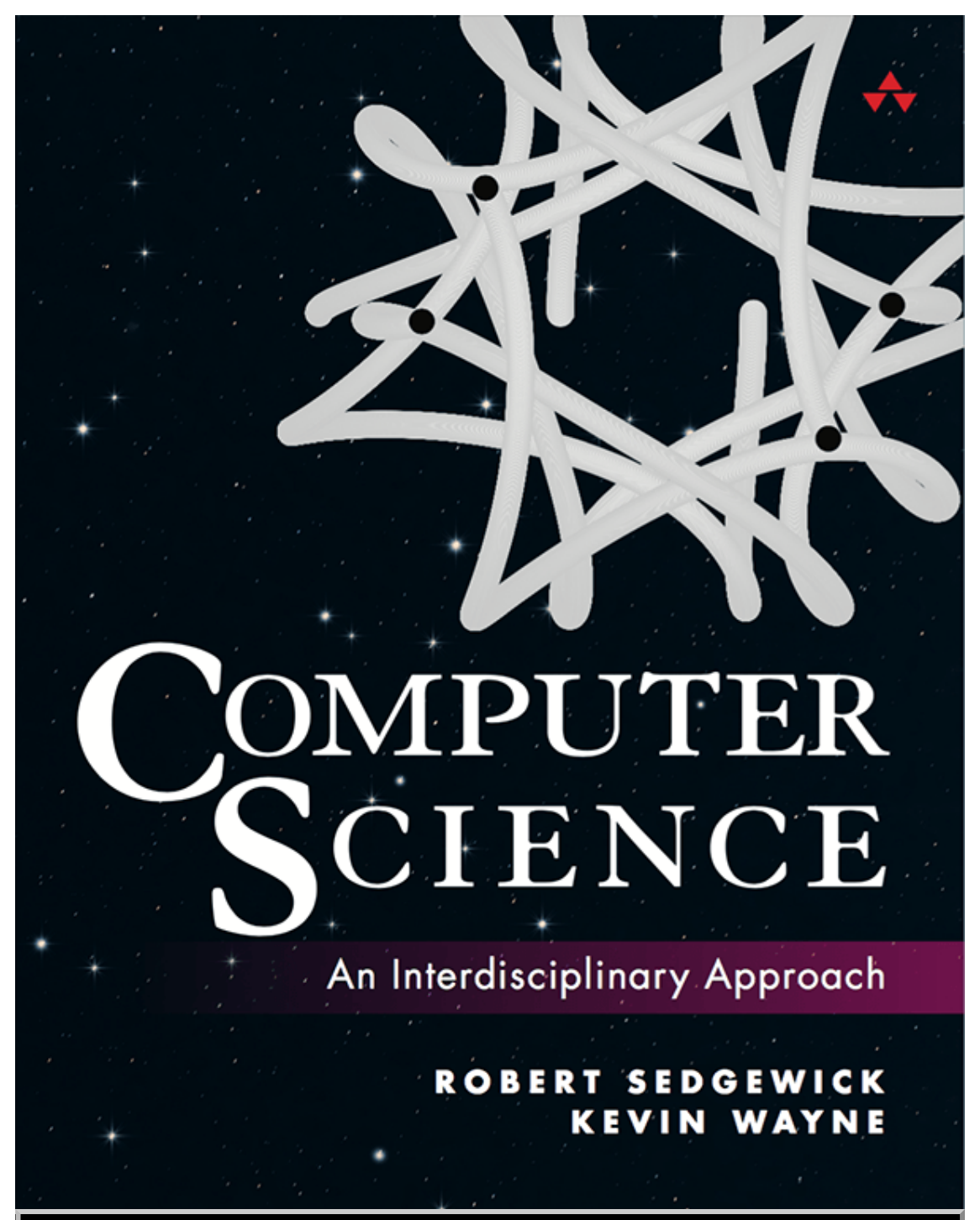
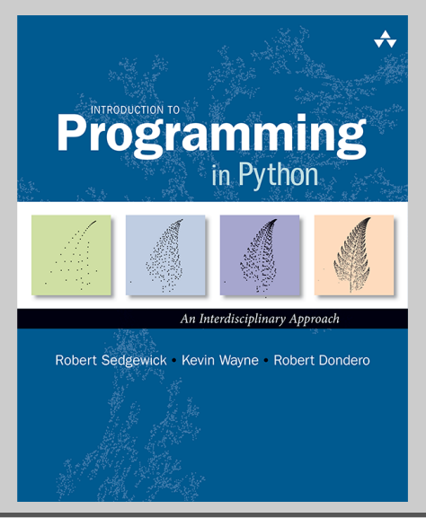
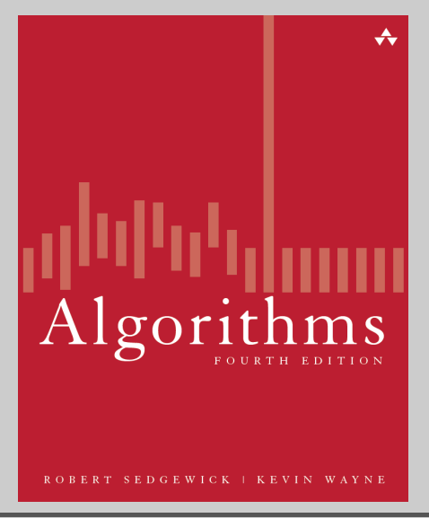


Intro to Programming
1. Elements of Programming
2. Functions
3. OOP
4. Data Structures
Computer Science
5. Theory of Computing
6. A Computing Machine
7. Building a Computer
Beyond
8. Systems
9. Scientific Computation
Related Booksites





Web Resources
FAQ
Data
Code
Errata
Lectures
Appendices
Online Course
Java Cheatsheet
Programming Assignments
ENHANCED BY Google

textbooks for a first course in computer science
for the next generation
of scientists and engineers

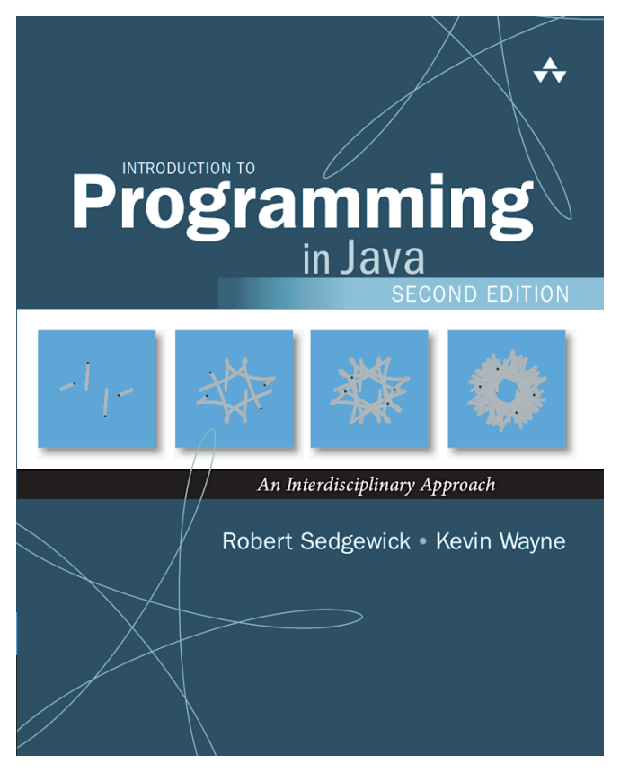
Online content. This booksite contains tens of thousands of files, fully coordinated with our textbook and also useful as a standalone resource. It consists of the following elements:

- *Excerpts.* A condensed version of the text narrative, for reference while online.
- *Lectures.* Curated studio-produced online videos, suitable for remote instruction.
- *Java code.* Hundreds of easily downloadable Java programs and our I/O libraries for processing text, graphics, and sound.
- *Data.* Real-world data sets for testing code (ours and yours).
- *Exercises.* Selected exercises from the book and “web exercises” developed since its publication, along with solutions to selected exercises.
- *Programming assignments.* Creative programming assignments that we have used at Princeton.

You can explore these resources via the sidebar at left.

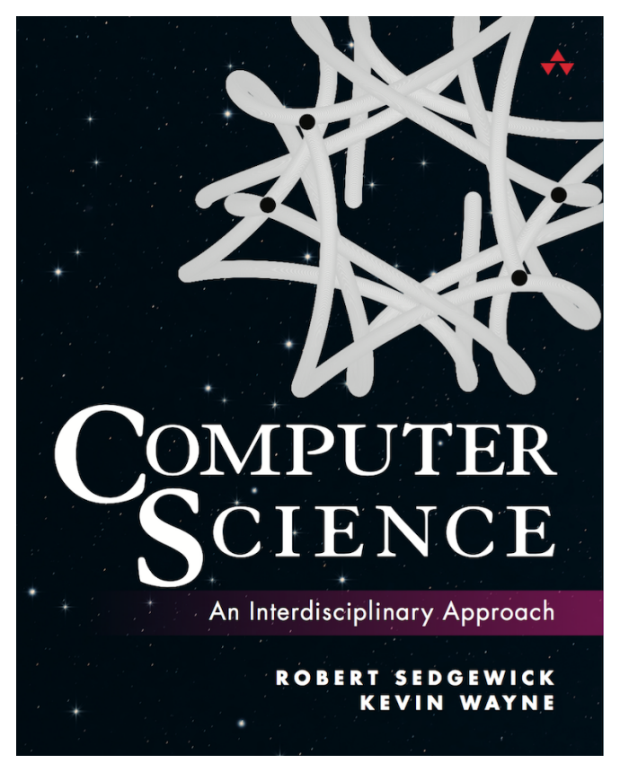
Introduction to Programming in Java. Our textbook *Introduction to Programming in Java* [Amazon · Pearson · InformIT] is an interdisciplinary approach to the traditional CS1 curriculum with Java. We teach the classic elements of programming, using an “objects-in-the-middle” approach that emphasizes data abstraction. We motivate each concept by examining its impact on specific applications, taken from fields ranging from materials science to genomics to astrophysics to internet commerce. The book is organized around four stages of learning to program:

- *Chapter 1: Elements of Programming* introduces variables; assignment statements; built-in types of data; conditionals and loops; arrays; and input/output, including graphics and sound.
- *Chapter 2: Functions* highlights the idea of dividing a program into components that can be independently debugged, maintained, and reused.
- *Chapter 3: Object-Oriented Programming* emphasizes the concept of a data type and its implementation, using Java's class mechanism.
- *Chapter 4: Algorithms and Data Structures* discusses classical algorithms for sorting and searching, and fundamental data structures, including stacks, queues, and symbol tables.



Computer Science. Our textbook *Computer Science* [Amazon · Pearson · InformIT] contains *Introduction to Programming in Java* as its first four chapters. The second half of the book explores core ideas of Turing, von Neumann, Shannon, and others that ignited the digital age.

- *Chapter 5: Theory of Computing* surveys the fundamental concepts of universality, computability, and intractability, which raise questions about the role of computation in understanding the natural world.
- *Chapter 6: A Computing Machine* describes a simple imaginary machine that has many of the characteristics of real processors at the heart of the computational devices that surround us.
- *Chapter 7: Building a Computer* considers the design of a processor, including Boolean logic, combinational circuits, and sequential circuits.



Reading a book and surfing the web are two different activities: This booksite is intended for your use while online (for example, while programming and while browsing the web); the textbook is for your use when initially learning new material and when reinforcing your understanding of that material (for example, when reviewing for an exam).

For teachers:

- *This online content.* Everything on these pages is freely available. We ask only that you adhere to normal academic traditions of attribution if you adapt this content in your own course. One best practice is to just provide links to our pages.
- *To use the lecture videos.* Please go to the Lectures tab at left for links to all the online videos and suggestions on how to use them.
- *To adopt the textbook.* You can request an examination copy or email the authors ✉ for more information. ACM/IEEE cites COS 126 as a course exemplar. Lecture slides (in Keynote format) are available by request ✉ for instructors who adopt the textbook.

For students:

- *This online content.* Whether your course uses our book or not, you can reinforce your understanding of many topics related to the study of algorithms by browsing the excerpts, code, and exercises here, watching the lecture videos, and/or using our book as a reference.
- *Java code.* Please go to the Code tab at left for instructions on setting up a Java programming environment, installing our standard libraries, and downloading all of our code.
- *Lecture videos.* Please go to the Lectures tab at left for links to all the online videos and suggestions on how to use them.
- *Online course.* You can take our free Coursera MOOCs Computer Science: Programming with a Purpose ∞ and Computer Science: Algorithms, Theory, and Machines ∞.

Python. We also have a Python version of Chapters 1–4 of our textbook.

Last modified on April 19, 2021.