教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

JAVA

—— 计算机科学与程序设计导论

(第二版 影印版)

JAVA

An Introduction to Computer Science and Programming

(Second Edition)

■ Walter Savitch

PEARSON
Prentice Hall

高等教育出版社
Higher Education Press

教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

JAVA

——计算机科学与程序设计导论

（第二版　影印版）

JAVA

An Introduction to Computer Science and Programming

(Second Edition)

Walter Savitch

高等教育出版社

图字：01-2003-5108 号

JAVA: An Introduction to Computer Science and Programming, Second Edition
Walter Savitch

原版 ISBN: 0-13-031697-0

图书在版编目（CIP）数据

JAVA：计算机科学与程序设计导论＝JAVA：An Introduction to Computer Science and Programming：第 2 版 /（美）萨维查（Savitch，W.）著．—影印本．—北京：高等教育出版社，2003.10
ISBN 7-04-013725-9

Ⅰ.J… Ⅱ.萨… Ⅲ.JAVA 语言—程序设计—英文
Ⅳ.TP312

中国版本图书馆 CIP 数据核字（2003）第 087886 号

出版发行	高等教育出版社	**购书热线**	010-64054588
社　　址	北京市西城区德外大街 4 号	**免费咨询**	800-810-0598
邮政编码	100011	**网　　址**	http://www.hep.edu.cn
传　　真	010-64014048		http://www.hep.com.cn
经　　销	新华书店北京发行所		
印　　刷	北京外文印刷厂		
开　　本	787×1092　1/16	**版　　次**	2003 年 10 月第 1 版
印　　张	68	**印　　次**	2003 年 10 月第 1 次印刷
字　　数	1 300 000	**定　　价**	68.00 元（附光盘）

本书如有缺页、倒页、脱页等质量问题，请到所购图书销售部门联系调换。

前　言

20 世纪末，以计算机和通信技术为代表的信息科学和技术对世界经济、科技、军事、教育和文化等产生了深刻影响。信息科学技术的迅速普及和应用，带动了世界范围信息产业的蓬勃发展，为许多国家带来了丰厚的回报。

进入 21 世纪，尤其随着我国加入 WTO，信息产业的国际竞争将更加激烈。我国信息产业虽然在 20 世纪末取得了迅猛发展，但与发达国家相比，甚至与印度、爱尔兰等国家相比，还有很大差距。国家信息化的发展速度和信息产业的国际竞争能力，最终都将取决于信息科学技术人才的质量和数量。引进国外信息科学和技术优秀教材，在有条件的学校推动开展英语授课或双语教学，是教育部为加快培养大批高质量的信息技术人才采取的一项重要举措。

为此，教育部要求由高等教育出版社首先开展信息科学和技术教材的引进试点工作。同时提出了两点要求，一是要高水平，二是要低价格。在高等教育出版社和信息科学技术引进教材专家组的努力下，经过比较短的时间，第一批引进的 20 多种教材已经陆续出版。这套教材出版后受到了广泛的好评，其中有不少是世界信息科学技术领域著名专家、教授的经典之作和反映信息科学技术最新进展的优秀作品，代表了目前世界信息科学技术教育的一流水平，而且价格也是最优惠的，与国内同类自编教材相当。

这项教材引进工作是在教育部高等教育司和高教社的共同组织下，由国内信息科学技术领域的专家、教授广泛参与，在对大量国外教材进行多次遴选的基础上，参考了国内和国外著名大学相关专业的课程设置进行系统引进的。其中，John Wiley 公司出版的贝尔实验室信息科学研究中心副总裁 Silberschatz 教授的经典著作《操作系统概念》，是我们经过反复谈判，做了很多努力才得以引进的。William Stallings 先生曾编写了在美国深受欢迎的信息科学技术系列教材，其中有多种教材获得过美国教材和学术著作者协会颁发的计算机科学与工程教材奖，这批引进教材中就有他的两本著作。留美中国学者 Jiawei Han 先生的《数据挖掘》是该领域中具有里程碑意义的著作。由达特茅斯学院 Thomas Cormen 和麻省理工学院、哥伦比亚大学的几

位学者共同编著的经典著作《算法导论》，在经历了11年的锤炼之后于2001年出版了第二版。目前任教于美国Massachusetts大学的James Kurose教授，曾在美国三所高校先后10次获得杰出教师或杰出教学奖，由他主编的《计算机网络》出版后，以其体系新颖、内容先进而倍受欢迎。在努力降低引进教材售价方面，高等教育出版社做了大量和细致的工作。这套引进的教材体现了权威性、系统性、先进性和经济性等特点。

教育部也希望国内和国外的出版商积极参与此项工作，共同促进中国信息技术教育和信息产业的发展。我们在与外商的谈判工作中，不仅要坚定不移地引进国外最优秀的教材，而且还要千方百计地将版权转让费降下来，要让引进教材的价格与国内自编教材相当，让广大教师和学生负担得起。中国的教育市场巨大，外国出版公司和国内出版社要通过扩大发行数量取得效益。

在引进教材的同时，我们还应做好消化吸收，注意学习国外先进的教学思想和教学方法，提高自编教材的水平，使我们的教学和教材在内容体系上，在理论与实践的结合上，在培养学生的动手能力上能有较大的突破和创新。

目前，教育部正在全国35所高校推动示范性软件学院的建设和实施，这也是加快培养信息科学技术人才的重要举措之一。示范性软件学院要立足于培养具有国际竞争力的实用性软件人才，与国外知名高校或著名企业合作办学，以国内外著名IT企业为实践教学基地，聘请国内外知名教授和软件专家授课，还要率先使用引进教材开展教学。

我们希望通过这些举措，能在较短的时间，为我国培养一大批高质量的信息技术人才，提高我国软件人才的国际竞争力，促进我国信息产业的快速发展，加快推动国家信息化进程，进而带动整个国民经济的跨越式发展。

教育部高等教育司

二〇〇二年三月

PREFACE FOR STUDENTS

This book is designed to teach you the Java programming language, and even more importantly, to teach you basic programming techniques. This book requires no previous programming experience and no mathematics other than some very simple high school algebra. However, to get the full benefit of the book, you should have a version of Java available on your computer, so that you can practice with the examples and techniques given in the book. You should have a version of Java called Java 2 (or some number higher than 2). If you have a version number of the form 1.1.x or 1.2.x, then the version number should be 1.2.x or higher. (The exact number that is filled in for the x is not critical. The x need not even be present. If it says only "version 1.2," that is fine.)

If You Have Programmed Before

You need not have any previous programming experience to use this book. This book was designed for beginners. However, the book can still be used to learn Java if you happen to have had experience with some other programming languages, but allow me to give you a few words of advice. If you have programmed before, do not assume that Java is the same as the programming language(s) you are used to using. All languages are different. And the differences, even if small, are large enough to give you problems. Read at least the boxed sections of Section 1.4 in Chapter 1 and all the boxed sections of Chapters 2 and 3. By the time you reach Chapter 4, it would be wise to read the entire chapter.

If you have programmed before in either C or C++, the transition to Java can be troublesome. While Java is very different from C and C++, at first glance it looks as if it is the same as C++. Appendix 11 has a comparison of Java and C++ that will help you see the differences between Java and C++ (or Java and C).

Copies of the Programs from the Text

This book contains a CD that includes all the programs and other software examples in the book, so that you can practice with these examples without having to type them into your computer.

Obtaining a Copy of Java

How and what version of Java you use depends somewhat on what operating system you are using. Be sure to consult the subsection below that corresponds to your operating system.

Microsoft Windows

Alternative 1:
The CD that comes with this book includes a version of JBuilder 3.5 Foundation, a complete Java integrated environment from Inprise/Borland. JBuilder includes an

editor and other utilities in addition to the Java language. This has everything you need to write and run Java programs. This is a professional strength environment, which can be a bit complex for novices, so we also have an alternative that gives you an easier environment.

Alternative 2:
This is a bit more complicated to initially set up, but easier to use once you do set up things. Download a free Java compile over the Internet from Sun Microsystems. Install that Java compiler and the TextPad environment, which is provided on the CD that comes with this book. The TextPad environment provides an editor and other tools to use when writing Java programs.

At the time this book went to press, the site for the Java compiler download from Sun Microsystems was:

```
http://java.sun.com/products/jdk/1.2/
```

Mac Operating System

There is a version of Java for the Mac that can be downloaded from the Sun Microsystems website. Unfortunately, users have not been happy with the Mac version of Java provided at this site, and indeed, it may not do all things discussed in this book.

If you are using the Mac operating system, one good alternative is to purchase a version of CodeWarrior from Metrowerks, Inc. It works well with the Mac operating system.

A version of JBuilder for the Mac is due out soon and promises be an excellent alternative for Mac users. You may want to check the following website to see if it is available. If it is, you can download it from there.

```
http://www.borland.com/jbuilder/foundation/download/
```

UNIX Operating System

Alternative 1:
The CD that comes with this book includes a version of JBuilder 3.5 Foundation, a complete Java integrated environment from Inprise/Borland. JBuilder includes an editor and other utilities in addition to the Java language. This has all the software you need in order to write and run Java programs. JBuilder has versions for both the Solaris and Linux operating systems.

Alternative 2:
You can down load a free Java compiler over the Internet from Sun Microsystems. At the time this book went to press, the site for the Java download from Sun Microsystems was:

```
http://java.sun.com/products/jdk/1.2/
```

We do not have an editor/environment (other than JBuilder) that we recommend for use with this compiler. You can use your favorite editor to write programs and then

run your Java programs from the command line as described in Chapter 1. (Or you may find an environment you like and can use it.)

Self-Test Questions

Each chapter contains numerous self-test questions. Complete answers for all the self-test questions are given at the end of each chapter. One of the best ways to practice what you are learning is to do the self-test questions *without looking at the answers.* Only look at the answers after you have answered the self-test questions.

This Text Is Also a Reference Book

In addition to using this book as a textbook, you can and should use it as a reference. When you need to check a particular point that you may have forgotten or that you hear mentioned by somebody but have not yet learned yourself, just look in the index. Many index entries give a page number for "quick reference." Turn to this quick reference page. It will contain a short entry, usually set off in a box, that gives all the essential points on that topic. This can be done to check details of the Java language, as well as details on programming techniques.

Boxed sections in every chapter give you a quick summary of the main points in that chapter. You can use these boxes to review the chapter, preview the chapter, or check details of the Java language.

Updates and Corrections

Any updates or corrections will be listed on the author's website for this book

```
http://www.cse.ucsd.edu/users/savitch/books/cs1.java/
```

We Want Your Opinions

This book was written for you, and I would like to hear any comments you have on the book. You can contact me via electronic mail at the following address:

```
wsavitch@ucsd.edu
```

Unfortunately, I cannot provide you with answers to the programming exercises. Only instructors who adopt the book can receive (selected) answers from the publisher. For help on the programming exercises, you will have to contact your instructor. (Even if you are not enrolled in a class we still cannot provide answers to programming exercises.) But, remember that there are answers to all the self-test questions at the end of each chapter.

Walter Savitch
http://www.cse.ucsd.edu/users/savitch

PREFACE FOR INSTRUCTORS

This book was designed to be used in a first course in programming and computer science. It covers programming techniques, as well as the basics of the Java programming language. It is suitable for courses as short as one quarter or as long as one full academic year. It requires no previous programming experience and no mathematics other than a little high school algebra. This book can also be used for a course designed to teach Java to students who have already had another programming course, in which case, the first few chapters can be assigned as outside reading. (If students have had previous programming experience in C or C++, then there is also an appendix that explains some differences between Java and C or C++.) All the code in the book has been tested using Java 2 of Sun Microsystems. The coverage of Java was carefully arrived at by class testing and is a concise, accessible introduction for beginners.

Changes in this Edition

If you have not used the first edition of this text, you can skip this subsection. If you have used the first edition, this subsection will tell you how this second edition differs from the first edition.

For instructors, the transition from the first edition of this text to this edition is easy. You can teach the same course with basically the same topics presented in the same order. Some chapters have changed numbers, but you can still cover those chapters in the order you are currently using. The biggest change was to move the arrays chapter forward to Chapter 6. However, you can cover arrays later if you prefer with no loss of continuity in reading the text. The only significant change you will need to contend with is that this edition uses the Swing library instead of using only the AWT library as the first edition did. However, there have been changes and additions that you may find helpful.

This edition adds coverage of the Swing Libraries, the `Graphics` class, and linked data structures to the topics covered in the first edition. In addition, the entire book has been rewritten to make the material clearer and more complete. There are many more Self-Test Questions and many more Programming Exercises in this edition.

In response to requests from users of the first edition, we have adopted the policy of listing instance variables first in class definitions (as opposed to last, as in the first edition).

This book also contains some early, optional material on applets and another GUI class named `JOptionPane`. This allows instructors to introduce GUI interfaces early if they wish, or wait to introduce them later (or not at all) if that is preferred.

Java 2 Coverage

The first edition of this book was already fully compatible with Java 2. This edition adds coverage of Swing and other Java 2 details to provide more complete coverage of Java 2.

Flexible

If you are an instructor, this book adapts to the way you teach, rather than making you adapt to the book. This book does not tightly prescribe the order in which your course must cover topics. Neither does it prescribe the specialized libraries that must be used in your course. You can easily change the order in which chapters and sections are covered. The details about rearranging material are explained in a chart at the end of this preface and in more details in a prerequisite section at the start of each chapter.

Since Java does not include any simple console input, most texts, even more advanced texts, provide some added class library for console input. This book requires that you add as little nonstandard software as possible, since only one simple class is added (for console input). Even that one console input class, which is included early in the book, becomes an understandable programming example for students well before the end of the book. All the remaining software is from standard Java libraries that should be part of any Java installation.

Coverage of Problem Solving and Programming Techniques

This book is designed to teach students basic problem-solving and programming techniques and is not simply a Java syntax book. The book contains numerous case studies and programming tips, as well as many other sections that explain important problem-solving and programming techniques, such as loop design techniques, debugging techniques, style techniques, abstract data types, basic object-oriented programming including event-driven programming, and other computer science topics.

Object-Oriented and Traditional Techniques

Any course that really teaches Java must teach classes early, since almost everything in Java involves classes. The behavior of parameters depends on whether they are class parameters. Even the behavior of the equals operator (==) depends on whether it is comparing objects or simpler data items. Classes cannot be avoided, except by means of absurdly long and complicated "magic formulas." This book introduces classes fairly early. Some exposure to using classes is introduced in Chapters 1 and 2. Defining classes is covered in Chapter 4. Moreover, all the basic information about classes, including inheritance, is presented by the end of Chapter 7 (and this can be done omitting Chapter 6). However, some topics on classes, including inheritance, can be postponed to later in a course.

Although this is an early classes book, it does not neglect traditional programming techniques, such as top-down design and loop design techniques. These older topics may no longer be glamorous, but they are information that all beginning students need.

Swing, Applets, and Other GUIs

Starting with Java 2, Java comes with an improved GUI library known as Swing that allows programmers to design portability GUIs (graphical user interfaces). This book uses Swing to teach students to produce professional looking windowing interfaces. In the process, students learn event-driven programming, as well as receiving a lot of practice with object-oriented programming.

As this material was class-tested and views of instructors were gathered, we found that Swing was a more accessible way to teach students object-oriented programming than applets. Thus, we place greater emphasis on Swing. This makes sense, since almost all advanced applets tools are really Swing tools. However, for those who do want to cover applets early, Chapter 1 has an optional section that previews applets. Chapter 13 covers applets in detail and may be covered much earlier than the chapter number suggests. You may choose to introduce GUIs early, late, or not at all.

With the introduction of the Swing libraries, there is a new class named `JOptionPane` that allows an easier introduction to GUIs than applets provide. This book covers `JOptionPane` in an optional section of Chapter 2. You have the choice of introducing either or both applets and `JOptionPane` either late or early (or not at all).

In addition to this optional GUI material in Chapters 1 and 2, this book includes three full chapters on GUIs, which gives thorough coverage of Swing, applets, and the `Graphics` class for simple two-dimensional graphics.

Language Details and Sample Code

This book teaches programming technique and does not simply teach the Java language. However, neither students nor instructors would be satisfied with an introductory programming course that did not also teach the programming language. Until you calm a student's fears about language details, it is often impossible to get her or his attention to discuss bigger issues. For this reason, this book gives complete explanations of Java language features and lots of sample code. Programs are given in their entirety along with sample input and output. In many cases, there are even extra complete examples on the CD, in addition to the complete examples in the text.

Self-Test Questions

Self-test questions are spread throughout each chapter. These questions have a wide range of difficulty levels. Some require only a one-word answer, whereas others require the reader to write an entire, nontrivial program. Complete answers for all

the self-test questions, including those requiring full programs, are given at the end of each chapter.

Class Tested

The material in this book has been fully class tested. Much of the material and methods of presentation were revised in response to this class testing.

Support Material

The support materials described below can be obtained from the publisher, obtained over the Internet, or are included with the book.

CD-ROM

Each book contains a CD that includes all the programs and classes in the book. The CD also includes a version of JBuilder 3.5 Foundation, a complete Java integrated environment from Inprise/Borland. JBuilder includes an editor and other utilities in addition to the Java language. The CD includes versions of JBuilder for Windows, Solaris, and Linux operating systems. The CD also includes a copy of TextPad, a very nice integrated environment that runs under Windows and that is a suitable environment for use with Sun's Java 2.

Free Software

You have a wide choice of free software to use with this book. As already noted the CD that comes with this book includes JBuilder and TextPad.

JBuilder works under Windows, Solaris, and Linux operating systems. At the time this book went to press a version of JBuilder for the Mac was not available, but was due out soon. You may want to check the following website to see if it is available. If it is, you can download it from there.

```
http://www.borland.com/jbuilder/foundation/download/
```

Another very good alternative is to download a version of Java-2 from Sun Microsystems website. At the time this book went to press, the URL for the website was:

```
http://java.sun.com/products/jdk/1.2/
```

Java-2 is the main version of Java that we used in developing this text. The TextPad environment (which comes on the CD that accompanies this book) is a good environment to use with Sun's Java-2, provided you are using a Windows operating system. TextPad only runs under Windows.

Instructor's Resource Guide and Companion Website

Instructor tools include a chapter-by-chapter Instructor's Resource Guide that contains numerous teaching hints, quiz questions with solutions, and solutions to many programming exercises. The Companion Website includes code, PowerPoint slides, and other teaching resources. Instructors should contact their Prentice Hall

sales representative to obtain a copy of the Instructor's Resource Guide and receive information on how to access the Companion Website. For the name and number of your sales representative, please call Prentice Hall Faculty Services at 1-800-526-0485. Additional information on this book and other Prentice Hall products can be found on Prentice Hall's website at

```
http://www.prenhall.com/
```

Updates and Corrections

Any updates or corrections will be listed on the author's website for this book

```
http://www.cse.ucsd.edu/users/savitch/books/cs1.java/
```

Acknowledgments

I thank the Computer Science and Engineering Department of the University of California, San Diego (UCSD), which is my home department and the place that I tested much of this material. Many students in my classes were kind enough to help correct preliminary versions of this text. These student comments and the comments of instructors who class tested this book were a tremendous help in shaping the final book. In particular, I extend a special thanks to Carole McNamee of California State University, Sacramento and to both Paul Kube and Susan Marx of UCSD; their feedback and class testing of earlier editions or drafts of the book was a great help to me in producing this edition.

I thank all the reviewers who took the time to read drafts of this or the previous edition of this book. They provided invaluable detailed comments and suggestions. In alphabetical order within each group, they are

Reviewers for this second edition:

Jim Buffenbarger—Idaho State
Martin Chetlen—Moorpark C.C.
Tom Cortina—SUNY, Stony Brook
Prasun Dewan—University of North Carolina
Laird Dornan—Sun Microsystems
H.E. Dunsmore—Purdue, Lafayette
Adel Elmaghraby—University of Louisville
Gopal Gupta—New Mexico State
Ric Heishman—North Virginia C.C.
Rob Kelly—SUNY, Stony Brook
Dr. Le Gruenwald—University of Oklahoma
Blayne Mayfield—Oklahoma State
Alan Saleski—Loyola, Chicago

Reviewers for the first edition:

Michael Clancy—University of California, Berkeley
Michael Godfrey—Cornell University
Robert Herrmann—Sun Microsystems, Java Soft

Robert Holloway—University of Wisconsin, Madison
Lily Hou—Carnegie-Mellon University
John Motil—California State University, Northridge
James Roberts—Carnegie-Mellon University
Nan C. Schaller—Rochester Institute of Technology
Ryan Shoemaker—Sun Microsystems, Inc.
Donald E. Smith—Rutgers University

I also thank all the individuals at Prentice Hall who organized the reviewing and production of this book. In particular, I thank Jake Warde for a masterful job of coordinating the entire processes including the reviews, Jerry Ralya my developmental editor for his excellent work in all aspects of the writing on this edition, Toni Holm for her work in coordinating things between offices, and to Gail Cocker, Heather Scott, and especially Scott Disanno for work on the design and production of the book. All these wonderful people cheerfully did a great job. I extend a special thanks to my publisher Alan Apt for his invaluable support and advice throughout the writing and production process.

I thank Lew Rakocy for his excellent work on the programming exercises added to this edition. I thank Brian Durney for his fine work on the instructor's support material.

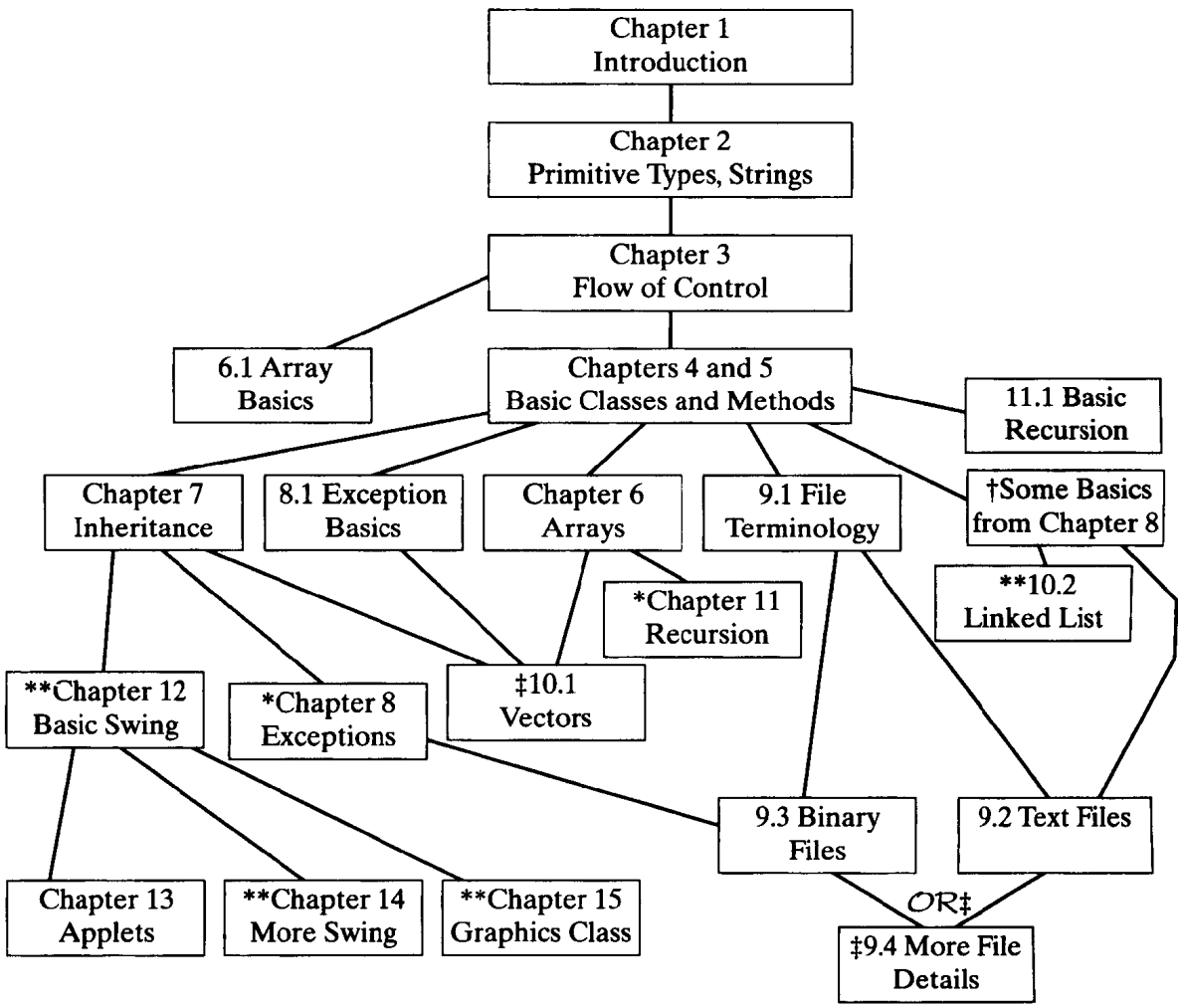
I thank Sun Microsystems for allowing me to use the Duke icon in a number of my GUI examples.

Finally, I give an extra special thanks to Christina for putting up with me while I worked late on this book and for even going so far as to proofread some sections of the book for me.

Walter Savitch
wsavitch@ucsd.edu
http://www.cse.ucsd.edu/users/savitch

DEPENDENCY CHART

If there is a line between two boxes, then the material in the higher box should be done before the material in the lower box. Minor variations to this chart are discussed in the prerequisites section at the start of each chapter. These variations usually provide more, rather than less, flexibility.



* Note that some sections of these chapters can be covered sooner. Those sections are given in this chart.

**See the chapter prerequisites section for full details.

† Section 8.1 and the subsection ***Declaring Exceptions*** of Section 8.3.

‡ 9.4 requires either text file or binary files, but not both.
Most of 10.1 (vectors) can be covered before covering inheritance.

TABLE OF CONTENTS