Explorations in Computer Science
SECOND EDITION
R. Mark Meyer
Canisius College
# Table of Contents

Preface .......................... vii
Acknowledgments ........................ x

**Laboratory 1: Introduction to the Labs** .......................... 1
   Applet: a) Introduction

**Laboratory 2: Exploring Number Systems** .......................... 9
   Applets: a) Number systems
            b) Binary addition

**Laboratory 3A: Representing Numbers** .......................... 19
   Applets: a) Negative binary numbers
            b) Real number representations

**Laboratory 3B: Colorful Characters** .......................... 29
   Applets: a) Character codes
            b) Text translator into ASCII
            c) Color maker

**Laboratory 3C: Compressing Text** .......................... 41
   Applets: a) Text compression using key words
            b) Text compression using
               Huffman encoding

**Laboratory 4: Logic Circuits** .......................... 51
   Applet: a) LogicGates

**Laboratory 5: Computer Cycling** .......................... 67
   Applet: a) Super Simple CPU

**Laboratory 6: Problem Solving** .......................... 81

**Laboratory 7: Low-Level Languages** .......................... 91
   Applet: a) Super Simple CPU

**Laboratory 8: Using Algorithms for Painting** .......................... 103
   Applet: a) Palgo
Laboratory 9A: Searching for the Right Sort .......................... 119
    Applets: a) Stackqueue
             b) Trees

Laboratory 9B: Searching for the Right Sort .......................... 134
    Applets: a) Searching
             b) Sorting
             c) Palgo

Laboratory 10: Operating Systems .......................................... 145
    Applets: a) Placement of jobs in memory
             b) Scheduling of jobs

Laboratory 11: Disk Scheduling ............................................... 155
    Applet: a) Disk Scheduling

Laboratory 12A: Spreadsheets ................................................. 165

Laboratory 12B: Databases ...................................................... 187
    Applet: a) Simple SQL

Laboratory 13: Artificial Intelligence ...................................... 201
    Applets: a) Semantic networks
             b) Eliza therapist

Laboratory 14: Simulating Life & Heat .................................... 211
    Applets: a) Game of Life
             b) Heat transfer

Laboratory 15: Networking ..................................................... 225
    Applets: a) TCP/IP
             b) Network router

Laboratory 16A: First Steps in HTML ...................................... 245

Laboratory 16B: Linking & Images in HTML ............................... 255
Laboratory 17: Limits of Computing ................................. 267

Applets:
   a) Comparison of several functions
   b) Plotter
   c) Traveling Salesperson Problem
Welcome to Explorations in Computer Science. This lab manual accompanies the textbook Computer Science Illuminated, 2/e by Nell Dale and John Lewis, Jones and Bartlett Publishers, ©2004. The goal of this lab manual is to provide online, computer-based activities that will reinforce the concepts presented in the textbook.

The labs in this manual are arranged and numbered to follow the chapters of Computer Science Illuminated, 2/e, except for Lab 1, which introduces skills needed to work with this manual. In those cases where there is more than one lab for a given chapter in the textbook (for example, Labs 3a, 3b, and 3c), each lab focuses in-depth on a particular topic covered in the text. Teachers can assign all of the labs, or can pick and choose which ones to use in order to tailor the course to their needs and priorities.

Assumptions

Since Computer Science Illuminated, 2/e is targeted for CS-0 courses, the activities in this lab manual assume that students have the skills necessary for a course at this level. For instance, it is assumed that the students have no prior experience with programming, though they probably have basic skills with a computer. Therefore, this manual does not attempt to teach anyone how to turn on the computer, use a mouse, or drag a window. One of the essential skills that students may not yet have is how to take a picture of what is on their screen (often called a screenshot), and this is explained in the first lab.

Another assumption is that students using this lab manual are not taking the course solely to learn a particular programming language, but rather to be exposed to the wide variety of topics within the realm of Computer Science. This assumption mirrors that of the textbook. There are enough activities for students to get a good feeling for algorithms and programming, but these do not form the majority of the materials presented.

Since this lab manual refers to specific pages in the textbook, students should have their copies of Computer Science Illuminated 2/e with them as they work through the activities. Detailed explanations of the core topics, such as what job scheduling is and how it is used in an operating system, are not recapitulated in this manual. Rather, the students are referred to appropriate chapters and pages. The lab manual commences with an explanation of how to use the applet provided on the CD and how to interpret its behavior. In a few instances throughout the book, the author couldn’t help interjecting some historical or other material deemed of interest.
Organization of the Labs

Each lab begins with succinct Objectives, References, and Background sections that briefly describe what is to be learned, what applets are to be used, and where in the textbook to find further information. Then the Activity section describes the software and shows how to use it. Students are expected to work through the Activity section, which guides them through starting the applet, comparing what they see on their screens to the screenshots in the lab manual, and completing the activities. This will prepare them to complete the Exercise sections. The teacher may assign one or more of the exercises to be handed in later.

The Deeper Investigation section provides a stimulating coda for advanced students. A further activity is briefly described or a question posed. In most cases, the appropriate response would be a one-page written answer to the question. In other cases, the teacher may assign the task posed by the Deeper Investigation.

Software

One very important goal of this manual is that students must be able to do the activities on any computer. In order to meet this goal, we include a CD with the lab manual, which has nearly all the software in the form of Java applets that can be run on any modern web browser.

The minimal software requirement for the lab is a web browser (Netscape 4 or higher, Internet Explorer 4 or higher, Linux Konquerer) with Java enabled.

All the applets were written using Java 1.2 and AWT graphical user interface components for maximum compatibility and widest audience availability. See the section below if you have trouble running the Java applets on your PC.

For one of the labs accompanying Chapter 12, we chose to use the spreadsheet program Microsoft Excel. Excel is a widely established program, available on both Windows and Macintosh. Students who use Microsoft Works instead will find no difference in the Works formulas, as least as far as the simple exercises that this lab manual poses are concerned. Linux and Unix users can use compatible products such as OpenOffice or SUN’s StarOffice, which uses the same formulas and even read and write Excel files. In some cases, students may not have access to any of these programs and can either skip the exercise or adapt it to whatever spreadsheet program to which they do have access. (Alas, the Microsoft Access database program is not as universal as Microsoft Excel — in particular, there is no Macintosh version. This is why we chose to use an applet for relational databases.)

Getting Java to Work in Your Web Browser

The applets and applications in this lab manual are all written in Java. Consequently, you will need Java’s run-time environment (JRE) to run them. You will need the version that we used in Java 1.2, so you may need a newer JRE.

Applets are java programs that run “inside” a web browser. That is, when you click on an html file or visit a web page that has a Java applet in it, the Java applet should start running automatically. If it doesn’t, you will get a blank area where the applet would have been, and sometimes a message like “Applet not initiated.”

Several years ago, court battles between Microsoft (makers of Windows XP and the Internet Explorer Browser) and Sun Microsystmes (owner of Java) resulted in Java not being automatically included in Microsoft’s popular web browser, Microsoft
Internet Explorer (MSIE). However, anybody can install Java on their Windows PC and run Java applets. It’s just that you may need to do this step once after you get a new computer.

Here’s how to get Java applets to run. First, go to Java’s web site: www.java.com. When you reach that page, if you see a cartoon movie playing, then Java is installed and usable through your web browser already. (The movie changes every few months so describing it is pointless, though it is always humorous, colorful, and has some tinny music accompanying it.) You can also click on “Verify installation” on the left side of the page to make sure your browser can understand Java.

But if you need to install the Java JRE, click on the “Help” link for more information, which is immediately above “Verify installation.” You can “Manually download” and install the JRE, too. There’s a link on http://www.java.com.

The above steps are all you need to use this lab manual. More adventurous students may want to compile Java programs as well, in which case you need to download the development kit. Go to http://java.sun.com and click on J2SE download. (J2SE stands for Java version 2 Standard Edition, which is free.) You could choose either J2SE 5.0 or J2SE 1.4.2, although the Java code in this lab manual was tested using J2SE 1.4.2. Remember, this step is not necessary if you merely want to run existing applets and applications.

How to Run the Software

Students can run the software directly off the CD included with their lab manuals by selecting the lab they want from the index. Nine of the applets also exist as stand-alone Java applications. This allows students to run the software on their computers outside a web browser, enabling them to load and save data files. For security reasons, a Java applet that sits on a web page cannot load or save files, making it impossible for students to turn in their work in the form of electronic files. Therefore we use screenshots as a way for students to document their work on the applets. In some cases, the lab instructor might prefer that students run the software as a stand-alone application when available, and then save electronic files that they can turn in.

Remember that Java applets are quite universal and can be used on almost any platform: Windows (in all its versions), Macintoshes, Linux, Unix, and other systems. Java’s graphics are also standard so applets that are written on one platform usually look identical, or nearly identical, on widely different platforms. This cuts down on development time and costs, and assures that the largest possible number of students can use the software. However, applets are, of course, somewhat limited in what they can do in order to foil viruses, worms, and Trojan horses.

Like most textbooks and all computer software, this lab manual and its accompanying software are evolving creations. Software problems and design issues have been addressed in this new edition and will be addressed in future revisions. Comments, suggestions, and field spottings of perhaps not-rare-enough bugs are definitely welcome. We hope students enjoy the exercises as they wind their way through this vast and fascinating landscape called Computer Science!
Acknowledgments

Numerous people have motivated this work and also made it possible. The kind folks at Jones & Bartlett Publishers set me on the path to writing this while we were visiting at SIGCSE ’02 in Covington, Kentucky and discussing the new textbook *Computer Science Illuminated*. However, this conversation would not have happened but for Dr. Jeffrey McConnell, chair of the Computer Science Department at Canisius College, who steered me toward the Jones & Bartlett booth. I sincerely thank him for this nudge that set a vast machine in motion.

A huge debt of thanks goes to Kelly Bucheger, who wrote two of the labs and fixed others. He has taught the CS-0 lab at Canisius for over twelve years and is a popular teacher on campus. A saxophonist, composer, and writer on jazz by profession, he rounds out his days teaching Computer Science and web design at several area colleges, including Canisius College. While possessing all the other attributes of a great teacher, namely compassion, gentleness, clarity, and fairness, he brings an essential ingredient to the lab: humor. Kelly knows how to make the lab time fun while motivating learning. We all strive for this but he has accomplished it. Hopefully students across the country and the world who read this lab manual will feel this spirit of fun in their adventure into Computer Science!

R. Mark Meyer