Preface

This manual consists of a series of laboratory experiments to accompany the eighth edition of the text *Computer Science: An Overview*, by J. Glenn Brookshear. These experiments, written in the spirit of learning by doing, are designed to give students a working knowledge of the Java programming language. The manual is divided into twelve sessions, each of which is designed for a two-hour laboratory session accompanied by an instructor. Each session closes with problems that can be assigned as desired to give students additional experience outside of the closed laboratory environment. The first laboratory session covers the particulars of the programming environment that will be used in the following laboratory sessions. It is meant to be supplemented by the local institution with materials that explain site-specific details. These materials should cover topics such as system entry, the file system, and commands related to editing, compiling, and running Java programs.

Special Note to Students

Your preparation before each laboratory session is extremely important. At minimum your preparation should include reading the entire session and its associated experiments. Your goal should be to identify the purpose of each experiment within the laboratory session before the actual session begins. A good approach is to read each experiment and then ask yourself why that particular experiment was included at that specific point in the laboratory session.

You should approach each session in a spirit of experimentation. The laboratory activities are not designed to tell you every detail about the topics covered. Instead, they are designed to encourage you to experiment and discover. Once you adapt to this mode of learning, you will find that a computer installation offers an endless opportunity to explore and learn.

In most cases the laboratory activities involve experimenting with short programs. Your instructor will tell you how to gain access to this software.

Special Note to Instructors

Each laboratory session contains more experiments than your students may be able to complete in a two-hour period. This becomes more pronounced as the sessions progress into experiments that require the development of entire routines. You are encouraged to assign those experiments that emphasize the topics you wish to cover.

Many of the experiments consist of running an example program and then modifying it. To avoid the tedium of typing and to allow students more time for experimentation and reflection, the initial form of these programs can be stored in files and made available to the students. This source code and other material supporting this manual are available at the web site at http://www.aw.com/brookshear

To assist you in associating software units with their place in the laboratory, each unit has a coded identifier. Each identifier begins with the letters J meaning "Java Laboratory." Following this are two digits identifying the session in which that unit is used. Following this is either the letter E or P, which indicates whether the unit is associated with an experiment or a post-laboratory problem. The identifier closes with two digits indicating the particular experiment or problem. In those cases in which more than one software unit is associated with a single experiment or problem, the identifier has an additional letter indicating whether the software is the first, second, etc., unit associated with that experiment or problem. Thus, the software identified as J03E02A is the first software unit needed for Experiment 2 in Laboratory Session.
3. These identifiers are included throughout the manual so that students can use them as a means of accessing the software within the laboratory environment.

This lab manual is an adaptation of the C++ Laboratory Manual written by Mary Boelk.

M.L.M.
Contents

Session 1 Introduction to the Local Computer System and the Execution of Java Programs
  Gaining Access to the Machine 1.2
  High-Level Programming Languages 1.2
  Object-Oriented Programming 1.3
  The Program Preparation Process 1.3
  A Simple Java Program 1.8
  Post-Laboratory Problems 1.13

Session 2 Data Types and the while Statement
  Primitive Data Types 2.2
  Variables 2.2
  Writing Data to the Monitor 2.3
  Reading Data from the Keyboard 2.5
  Elementary Operations and the Assignment Statement 2.9
  The while Control Statement 2.16
  Post-Laboratory Problems 2.19

Session 3 Data Storage and Its Implications
  Data of Type Integer 3.2
  Data of Type Real 3.5
  Data of Type Character 3.8
  Conversions 3.10
  Post-Laboratory Problems 3.15

Session 4 Machine Language and Its Relationship to Java
  Machine Language 4.2
  Bytecode 4.2
  The Machine Simulator 4.2
  Machine-Level Instructions 4.5
  Self-Modifying Programs 4.8
  Machine Language Summary 4.12
  Post-Laboratory Problems 4.13

Session 5 Control Statements
  The while and do-while Statements 5.2
  Boolean Expressions 5.6
  The if and if-else Statements 5.9
  The for Statement 5.12
  The switch Statement 5.16
  Post-Laboratory Problems 5.18

Session 6 Using Objects from the Java API
  Understanding a Method Header 6.2
  Declaring, Creating and Using a JOptionPane Object 6.3
  Declaring, Creating and Using a Random Object 6.8
  Declaring, Creating and Using a String Objects 6.13
  Post-Laboratory Problems 6.24
Session 7 Fundamentals of Defining a Utility Class
  Writing a First Utility Class - Part 1 - The Basics  7.2
  Writing a Method  7.3
  Writing a First Utility Class - Part 2 - Writing get and set Methods  7.6
  Writing a First Utility Class - Part 3 - Writing a Constructor  7.10
  Writing a First Utility Class - Part 4 - More Constructors and Methods 7.11
  Class Variables and Class Methods in the Integer and Math Classes 7.18
  Post-Laboratory Problems  7.22

Session 8 More About Objects
  How Objects are Stored  8.2
  The equals method Versus the Equals Operator ==  8.6
  Passing Objects to Methods  8.13
  Post-Laboratory Problems 8.17

Session 9 Recursion
  Recursive Methods in Java  9.2
  Iteration Versus Recursion  9.7
  Post-Laboratory Problems  9.13

Session 10 Arrays
  One-Dimensional Arrays 10.2
  Arrays of Objects  10.12
  Explaining String[] args  10.15
  Arrays of Arrays (Multi-Dimensional Arrays) 10.17
  Post-Laboratory Problems 10.20

Session 11 Inheritance and Polymorphism
  Fundamentals of Inheritance - Part 1  11.2
  Fundamentals of Inheritance - Part 2  11.5
  Polymorphism and Inheritance  11.14
  The Object Class  11.18
  Abstract Classes and Interfaces  11.21
  Post-Laboratory Problems  11.24

Session 12 Applets
  Applet Basics  12.2
  A First Applet  12.3
  The Color Class  12.7
  Components, Events and Listeners in an Applet  12.14
  Post-Laboratory Problems 12.20