1

An Introduction to Computers and Problem Solving

1.1 An Introduction to Computers

An Introduction to Programming Using Visual Basic 2010 is a book about problem solving using computers. The programming language used is Visual Basic 2010 (hereafter shortened to Visual Basic), but the principles taught apply to many modern programming languages. The examples and exercises present a sampling of the ways that computers are used in the real world. Here are some questions that you might have about computers and programming.

Question: *How do we communicate with the computer?*

Answer: Many languages are used to communicate with the computer. At the lowest level, there is *machine language*, which is understood directly by the microprocessor but is awkward for humans. Visual Basic is an example of a *higher-level language*. It consists of instructions to which people can relate, such as Click, If, and Do.

Question: What is a compiler?

Answer: A compiler is a program that translates a high-level language such as Visual Basic into machine language. The Visual Basic compiler detects (and points out) certain types of errors during the translation process.

Question: What is a GUI?

Answer: GUI (pronounced GOO-ee) stands for "graphical user interface." Both Windows and Visual Basic use a graphical user interface; that is, they employ graphic objects such as buttons and menus to interact with the user. Non-GUI text-based programs were common before 1990 but are now quite rare.

Question: *How do we get computers to perform complicated tasks?*

Answer: Tasks are broken down into a sequence of instructions that can be expressed in a computer language. (This text uses the language Visual Basic.) This sequence of instructions is called a *program*. Programs can range in size from two or three instructions to millions of instructions. Instructions are typed on the keyboard or read in from a file on a disk and are stored in the computer's memory. The process of executing the instructions is called *running* the program.

Question: *Are there certain features that all programs have in common?*

Answer: Most programs do three things: take in data, manipulate them, and give desired information. These operations are referred to as *input*, *processing*, and *output*. The input data might be held in a portion of the program, reside on a disk drive, or be provided by the computer user in response to requests made by the computer while the program is running. The processing of the input data occurs inside the computer and can take from a fraction of a second to many hours. The output data are either displayed on the monitor, printed on the printer, or recorded on a disk. As a simple example, consider a program that computes sales tax. An item of input data is the cost of the thing purchased. The processing consists of multiplying the cost by a certain percentage. An item of output data is the resulting product, the amount of sales tax to be paid.

Question: *What are the meanings of the terms "hardware" and "software"?*

Answer: *Hardware* refers to the physical components of the computer, including all peripherals, the central processing unit, disk drives, and all mechanical and electrical devices. Programs are referred to as *software*.

Question: *What are the meanings of the terms "programmer" and "user"?*

Answer: A *programmer* is a person who solves problems by writing programs on a computer. After analyzing the problem and developing a plan for solving it, he or she writes and tests the program that instructs the computer how to carry out the plan. The program might be run many times, either by the programmer or by others. A *user* is any person who uses a program. While working through this text, you will function both as a programmer and as a user.

Question: *What is meant by problem solving?*

Answer: Problems are solved by carefully reading them to determine what data are given and what outputs are requested. Then a step-by-step procedure is devised to process the given data and produce the requested output. This procedure is called an *algorithm*.

Question: *How did Visual Basic 2010 evolve?*

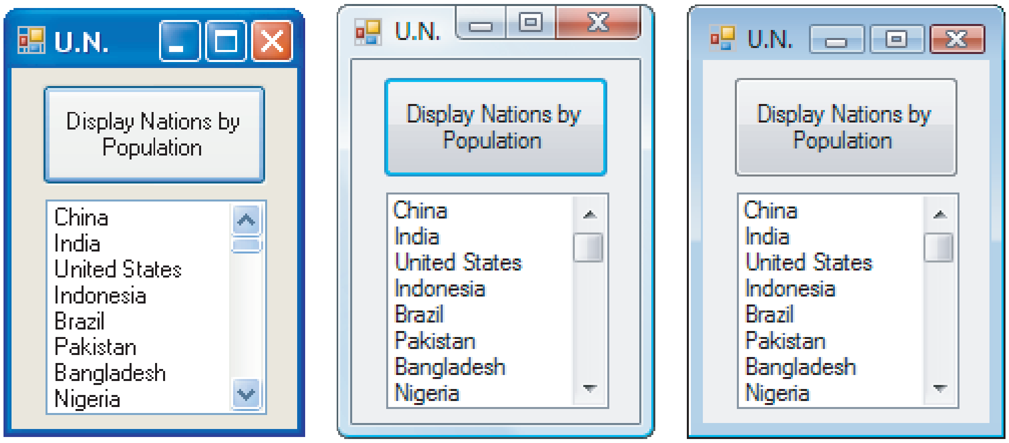
Answer: In the early 1960s, two mathematics professors at Dartmouth College developed BASIC to provide their students with an easily learned language that could tackle complicated programming projects. As the popularity of BASIC grew, refinements were introduced that permitted structured programming, which increased the reliability of programs. Visual Basic 1.0 is a version of BASIC developed in 1991 by the Microsoft Corporation to allow easy, visual-oriented development of Windows applications. Visual Basic 2010 is a language similar to the original Visual Basic, but more powerful.

Question: *Are there any prerequisites to learning Visual Basic 2010?*

Answer: Since Visual Basic is used to write Windows applications, you should be familiar with Windows and understand how folders and files are managed with Windows. The key concepts are presented succinctly in Section 1.2 and discussed in detail in Appendix C.

Question: *Will it matter whether I use Windows XP, Windows Vista, or Windows 7 as the operating system?*

Answer: Visual Basic runs fine with all three versions of Windows. However, the windows will vary in appearance. Figures 1.1(a) and 1.1(b) show the appearance of a typical window produced in Visual Basic with Windows XP and Windows Vista. With Windows 7, the appearance of windows depends on the Windows 7 product edition (such as Starter, Home Basic, Home Premium, etc.), the hardware on your system, and your own personal preferences. If you are using Windows 7 Starter or Windows 7 Home Basic, most likely your windows will look like the one in Fig. 1.1(c). With higher-end versions of Window 7 and recent hardware, your windows most likely will have an appearance similar to Fig. 1.1(b), known as *Aero*. In this book, all windows have the Aero appearance. No matter which operating system and appearance is used, the size and placement of the items inside the window should be the same.



(a) Windows XP (b) Windows Vista (c) Windows 7

FIGURE 1.1 **A Visual Basic window.**