Information Technology

An Introduction for Today's Digital World

Richard Fox

CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

Information Technology

An Introduction for Today's Digital World

Richard Fox

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20131028

International Standard Book Number-13: 978-1-4665-6829-7 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

CHAPTER 1

Introduction to Information Technology

This textbook is an introduction to information technology (IT) intended for students in IT-related fields. This chapter introduces the different career roles of an IT person, with emphasis on system administration, and the types of skills required of an IT professional. In this chapter, the elements that make up the IT infrastructure—the computer, software, users—are introduced.

The learning objectives of this chapter are to

- Describe and differentiate between types of IT careers.
- Describe the set of skills required to succeed in IT.
- Introduce the types of hardware found in a computer system.
- Describe and differentiate between the components of a computer system: hardware, software, and users.

WHAT IS INFORMATION TECHNOLOGY?

So, what is information technology (IT) anyway? IT is a term used to describe several things, the task of gathering data and processing it into information, the ability to disseminate information using technology, the technology itself that permits these tasks, and the collection of people who are in charge of maintaining the IT infrastructure (the computers, the networks, the operating system). Generically, we will consider IT to be the technology used in creating, maintaining, and making information accessible. In other words, IT combines people with computing resources, software, data, and computer networks.

IT personnel, sometimes referred to collectively as "IT," are those people whose job it is to supply and support IT. These include computer engineers who design and build computer chips, computer scientists who write software for computers, and administrators who provide the IT infrastructure for organizations.

What will your role be in IT? There are many varied duties of IT personnel. In some cases, a single individual might be the entire IT staff for an organization, but in many cases, there will be several, perhaps dozens or hundreds of personnel involved, each with separate roles to play. Most IT personnel, however, have two general roles: administration and support. An administrator is someone who is in charge of some portion of the IT infrastructure. There are a variety of administrator roles, as shown in Table 1.1.

Let us examine some of the administrator roles in Table 1.1 in more detail. The most common role in IT is the system administrator. System administration is the process of maintaining the operating system of a computer system. On a stand-alone computer, system administration is minimal and usually left up to the individual(s) using the computer. However, for a network of computers or computers that share files or other resources, system administration becomes more significant and more challenging. The system administrator is the person (or people) who perform system administration.

Maintenance of a computer system (computers, resources, network) will include an understanding of software, hardware, and programming. From a software point of view, administration requires installing software, making it available, troubleshooting problems that arise during usage, and making sure that the software is running efficiently. Additionally, the administrator(s) must understand the operating system well enough to configure the software appropriately for the given organization, create accounts, and safeguard the system from outside attack.

From a hardware point of view, administration requires installing new hardware and troubleshooting existing hardware. This may or may not include low-level tasks such as repairing components and laying network cable. It may also require installing device driver software whenever new hardware is added.

From a programming point of view, operating systems require "fine-tuning," and thus administrators will often have to write their own shell scripts to accomplish both simple

TABLE 1.1 Administrator Roles in IT

Role	Job/Tasks
System Administrator	Administer the computers in an organization; install software; modify/update operating system; create accounts; train users; secure system; troubleshoot system; add hardware
Network Administrator	Purchase, configure, and connect computer network; maintain computer network; troubleshoot network; secure network from intrusion
Database Administrator	Install, configure, and maintain database and database management system; back up database; create accounts; train users
Web Administrator	Install, configure, and maintain website through web server; secure website; work with developers
Web Developer	Design and create web pages and scripts for web pages; maintain websites
Security Administrator	Install, configure, and administer firewall; create security policies; troubleshoot computer system (including network); work proactively against intrusions

and complex tasks. In Linux, for instance, many components of the operating system rely on configuration files. These are often shell scripts. An administrator may have to identify a configuration file and edit it to tailor how that component works within the organization. The goal of writing shell scripts is to automate processes so that, once written, the administrator can call upon the scripts to perform tasks that otherwise would be tedious. A simple example might be to write a script that would take a text file of user names and create a new account for each user name.

System administration may be limited to the administration of the computers, printers, and file servers. However, system administration may extend to network administration and possibly even web server, ftp server, mail server, and database server administration depending on the needs and size of the company and abilities of the system administrator(s). Finally, a system administrator may also be required to train users on the system. Therefore, the skills needed for system administration can vary greatly. Specific common tasks of a system administrator include:

- Account management: creating new user accounts and deleting obsolete user accounts.
- Password management: making sure that all users have passwords that agree with the security policy (e.g., passwords must be changed every month, passwords must include at least one non-alphabetic character)—you might be surprised, but in systems without adequate password management, many users use "" as their password (i.e., their password is just hitting the enter key). Most organizations today require the use of *strong passwords*: passwords that contain at least eight characters of which at least one is non-alphabetic and/or a combination of upper- and lower-case letters, and are changed at least once a month without reusing passwords for several months at a time.
- File protection management: making sure that files are appropriately protected (for instance, making sure that important documents are not writable by the outside world) and performing timely backups of the file system.
- Installing and configuring new hardware and troubleshooting hardware including the network.
- Installing and configuring new software including updating new operating system (OS) patches, and troubleshooting software.
- Providing documentation, support, and training for computer users.
- Performing system-level programming as necessary (usually through scripting languages rather than writing large-scale applications or systems software).
- Security: installing and maintaining a firewall, examining log files to see if there are any odd patterns of attempted logins, and examining suspicious processes that perhaps should not be running.

In many cases, the network administrator is separate from the system administrator. It is the network administrator who is in charge of all aspects of the computer network. The network administrator's duties will include physically laying down cable, making connections, and working with the network hardware (for instance, routers and switches). The network administrator will also have to configure the individual machines to be able to communicate via the network. Thus, like the system administrator, the network administrator will edit configuration files, install software (related to the network), and so forth. Troubleshooting the network will also be a role for the network administrator where, in this case, troubleshooting may combine physical troubleshooting (e.g., is a cable bad?) and software troubleshooting. There is also a security aspect to the computer network. Both the system administrator and network administrator may work on system firewalls. Editing configuration files, writing shell scripts, and installing software and patches will all be part of a network administrators tasks.

Aside from administrative tasks, IT personnel provide support. Support usually comes in two forms: training and help desk. By training, the IT person is responsible for teaching new and current users how to use the IT infrastructure. This may include such simple things as logging into the computer system, setting up printers, accessing shared files, and perhaps training employees in how to use work-related software. The person might create documentation, helpful websites (including wiki pages), and even audiovisual demos, or lead group or individualized training sessions. Because training occurs only as needed (new software, new employees), most support comes in the form of a help desk. In essence, this requires that someone be available to respond to problems that arise at random times. Many large organizations offer 24/7 help desks. The help desk person might simply act as a switchboard, routing the problem to the proper IT person. In other cases, the help desk person can solve the problem directly, often over the phone but sometimes by e-mail or in person.

Where is IT used? IT is ubiquitous today. Nearly everyone on the planet uses some form of computing technology through cellular phones and tablets or home computers, or through school and work. However, most IT personnel are hired to work in IT departments for organizations. These organizations can be just a few people or corporations of tens of thousands. Table 1.2 provides a look at the larger users of IT and how they use IT.

TABLE 1.2 Large-Scale IT Users

Type of Organization	Typical Usage
Business	E-commerce, customer records
Education	Scholastic record keeping, support of teaching
Entertainment	Digital editing, special effects, music composition, advertising
Government	Record keeping, intelligence analysis, dissemination of information
Health/hospitals	Record keeping, medical devices, insurance
Law enforcement	Record keeping, information gathering, and dissemination
Manufacturing	Design, automation/robotics
Research	Computation, dissemination of information

WHO STUDIES IT?

IT personnel in the past were often drafted into the position. Consider the following scenario. Joe received his bachelor's degree in Computer Science from the University of Illinois. He was immediately hired by a software firm in Chicago where he went to work as a COBOL programmer. However, within 3 months, he was asked by the boss, being the new guy, "surely you know something about this Linux operating system stuff, don't you?" Joe, of course, learned Unix as part of his undergraduate degree and answered "Sure." So the boss told Joe "From now on, I want you to spend 10 hours of your week putting together this new network of computers using Linux. Make sure it can connect to our file servers and make it secure." Joe spent 10 hours a week reading manuals, installing the Linux operating system, playing around with the operating system, and eventually getting the system up and running.

After some initial growing pains in using the system, more and more employees switched to the Linux platform. Now, 9 months later, half of the company has moved to Linux, but the system does not necessarily run smoothly. Whenever a problem arises, Joe is usually the person who has to respond and fix it. The boss returns to Joe and says "Fine work you did on the network. I want to move you full time to support the system." Joe did not go to school for this, but because he had some of the skills, and because he is an intelligent, hardworking individual (he would have to be to graduate from University of Illinois's Computer Science program!), he has been successful at this endeavor. Rather than hiring someone to maintain the system, the easier solution is to move Joe to the position permanently. Poor Joe, he wanted to write code (although perhaps not COBOL). But now, the only code he writes are Linux shell scripts!

Sound unlikely? Actually, it was a very common tale in the 1980s and 1990s and even into the 2000s. It was only in the mid 2000s that an IT curriculum was developed to match the roles of IT personnel. Otherwise, such jobs were often filled by computer scientists or by people who just happened to be computer hobbyists. The few "qualified" personnel were those who had associates degrees from 2-year technical colleges, but those colleges are geared more toward covering concepts such as PC repair and troubleshooting rather than system and network administration. Today, we expect to see IT people who have not only been trained on the current technology, but understand all aspects of IT infrastructure including theoretical issues, the mathematics of computers (binary), the roles of the various components that make up a computer system, programming techniques, the operations of databases, networks, the Internet, and perhaps specialized knowledge such as computer forensics.

Common IT curricula include introductions to operating system platforms, programming languages, and computing concepts. We would expect a student to have experience in both Windows and Linux (or Unix). Programming languages might include both scripting languages such as Linux/Unix shell scripting, Ruby or Python, and JavaScript, and compiled languages such as C, C++, Java, or Visual Basic. Concepts will include operating systems and networks but may go beyond these to include web infrastructure, computer architectures, software applications (e.g., business software), digital media and storage, and e-commerce.

TYPES OF IT PROGRAMS

Although the 4-year IT degree is relatively new, it is also not standardized. Different universities that offer such an IT program come at the degree from different perspectives. Here, we look at the more common approaches.

First are the programs that are offshoots of computer science degrees. It seems natural to couple the degrees together because there is a good deal of overlap in what the two disciplines must cover: hardware technology, programming, database design, computer ethics, networking. However, the computer science degree has always heavily revolved around programming, and the IT degree may require less of it. Additionally, math plays a significant role in computer science, but it is unclear whether that amount of math is required for IT.

Next, there are the management information systems variations. The idea is that IT should be taught from a usage perspective—more on the applications, the data storage, the database, and less on the technology underlying the business applications. E-commerce, database design, data mining, computer ethics, and law are promoted here. Furthermore, the course work may include concepts related to managing IT.

Then there is the engineering technology approach that concentrates on hardware—circuit boards, disk drives, PC construction and troubleshooting, physical aspects of networking. There is less emphasis on programming, although there is still a programming component.

Another school of thought is to provide the foundations of computer systems themselves. This textbook follows this idea by presenting first the hardware of the computer system and then the operating systems. We also look at computer networks, programming, and computer storage to have a well-rounded understanding of the technology side to IT. The IT graduate should be able to not only work on IT, say as an administrator, but also design IT systems architecturally from the hardware to the network to the software.

SIGITE, the ACM Special Interest Group on IT Education, provides useful guidelines to build a model IT curriculum.

Who should study IT? To be an IT person, you do not necessarily have to have the rigorous mathematical or engineering background of computer scientists and computer engineers; there are many overlapping talents. Perhaps the most important talent is to have *troubleshooting* skills. Much of being an IT person is figuring out what is going wrong in your system. These diagnostic skills cannot be purely taught. You must have experience, background knowledge, and instinct. Above all, you have to know how the system works whether the system is a Linux operating system, a computer network, a web server, or other. Another talent is the ability to write program code—in all likelihood, you would write small programs, or scripts, as opposed to the software engineer who will be involved in large-scale projects.

You should also be able to communicate with others so that you can understand the problems reported by your colleagues or clients, and in turn describe solutions to them. This interaction might take place over the phone rather than in person. You should also be able to write technically. You may often be asked to produce documentation and reports. Finally, you will need the ability to learn on your own as technology is ever-changing. What you have learned in school or through training may be obsolete within a year or two. Yet, what you learn should form a foundation from which you can continue to learn. See Table 1.3, which highlights the skills expected or desired from IT personnel.

TABLE 1.3 IT Skills

Skill	Description	Example(s)
Troubleshooting, problem solving	Detect a problem Diagnose its cause Find a solution (means of fixing it)	Poor processor performance Disk space full Virus or Trojan horse infection
Knowledge of operating systems	Operating system installation Application software installation User account creation System monitoring	Versions of Linux Versions of Unix Windows Mac OS
System level programming	Shell scripts to automate processes Manipulating configuration files for system services	Bash, Csh scripts DOS scripts Ruby scripts C/C++ programs
System security	Ensuring proper system security is in place Following or drafting policies for users Monitoring for threats	Configuring a system firewall Installing antiviral/antimalware software Examining log files for evidence of intrusion and system security holes Keeping up with the latest security patches
Hardware	Installing and configuring new hardware Troubleshooting, replacing or repairing defective hardware	Replacing CPUs and disk drives Connecting network cables to network hubs, switches, routers

There probably is not a prototypical IT student. But an IT student should:

1. Enjoy playing around with the computer—not just using it, but learning how it works, learning how to do things on it at the system level
2. Enjoy learning on your own—liking the challenge of figuring things out, especially new things
3. Think that technology is cool—to not be afraid of technology but to embrace it in all of its forms
4. Enjoy troubleshooting

It is not necessarily the case that the IT student will enjoy programming. In fact, many students who select IT as a career make this choice because they first start with computer science but soon tire of the heavy programming requirements of that discipline. This is not to say that the IT student does not program, but that the programming is less intensive, requiring mostly writing small shell scripts. As an example, a student of computer science might, in the course of his or her studies, write a word processor, a database management system, or a language translator, whereas an IT student might write scripts to automate user account creation, or write client-side scripts to ensure that web forms have been filled in correctly, or write server-side scripts to process web forms.

There are many facets of the system administration position not covered above that are worth noting. Students may think that by studying for an IT career, they will get a job where they get to "play around" with technology. It is fun, but it is also a challenge—it is almost something that they already do as a hobby. And yet the student, when hired, might be responsible for maintaining the IT infrastructure in an organization of dozens or hundreds of employees. The equipment may cost hundreds of thousands of dollars, but the business itself might make millions of dollars. Therefore, the IT specialist must take their job seriously—downtime, system errors, intrusions, and so forth could cost the organization greatly. The IT specialist has duties that go beyond just being a system administrator. Some of these expectations are elaborated upon below.

To start, the system administrator must be aware of developments in the field. At a minimum, the system administrator has to know the security problems that arise and how to protect against them. These might include securing the system from virus, network intrusions, denial of service attacks, and SQL injection attacks. In addition, the system administrator should keep up on new releases of the operating system and/or server software that he/she maintains. However, a system administrator may have to go well beyond by reading up on new hardware, new software, and other such developments in the field.

In order for the system administrator to keep up with the new technology, new trends, and new security fixes, continuing education is essential. The system administrator should be a life-long learner and a self-starter. The system administrator might look toward Internet forums but should also regularly read technology news and be willing to follow up on articles through their own research. The system administrator should also be willing to dive into new software and experiment with it to determine its potential use within the organization.

A system administrator will often be "on call" during off hours. When disaster strikes, the system administrator must be accessible. An emergency call at 3 A.M. or while you are on vacation is quite possible. Although every employee deserves their own down time, a system administrator's contract may include clauses about being reachable 24/7. Without such assurance, an organization may find themselves with inaccessible data files or the inability to perform transactions for several hours, which could result in millions of dollars of damage. Some companies' reputations have been harmed by denial of service attacks and the inability to recover quickly.

The system administrator must also behave ethically. However, it is often a surprise to students that ethics is even an issue. Yet, what would you do if you are faced with some moral dilemma? For instance, your employer is worried that too many employees are using company e-mail for personal things, and so the boss asks you to search through everyone's e-mail. How would you feel? Now, imagine there is a policy in the company that states that employees can use company e-mail for personal purposes as long as e-mail does not divulge any company secrets. In this case, if you are asked to search through employee e-mail, would this change how you feel about it?

Unethical behavior might include:

- Spying on others (e-mail, web browsing habits, examining files)
- Setting up backdoor accounts to illegally access computer systems
- Illegally downloading software or files, or encouraging/permitting others to do so
- Performing theft or sabotage because of your system administration access

IT INFRASTRUCTURE

IT revolves around the computer. Have you used a computer today? Even if you have not touched your desktop (or laptop) computer to check your e-mail, chances are that you have used a computer. Your cell phone is a computer as is your Kindle. These are far less powerful than desktop units, but they are computers nonetheless. There are computer components in your car and on the city streets that you drive. The building you work or study in might use computers to control the lighting and air conditioning. Yes, computers are all around us even if we do not recognize them.

We will define a computer to be a piece of electronic equipment that is capable of running programs, interacting with a user (via input–output devices), and storing data. These tasks are often referred to as the IPOS (input, processing, output, storage) cycle. A general-purpose computer is one that can run any program. Many devices today are computers but may not be as general purpose as others. For instance, your iPod is capable of playing music; it has a user interface, and may have a small number of applications loaded into it to handle a calendar, show you the time of day, and offer a few games. Your cell phone has an even greater number of applications, but it is not capable of running most software. The degree to which a computer is general purpose is largely based on its storage capacity and whether programs have been specifically compiled for the processor.

Computers range in size and capability—from supercomputers that can fill a room, to desktop units that are not very heavy but are not intended to be portable, to laptop units that are as light as perhaps a heavy textbook, to handheld devices such as cell phones and mp3 players. The general difference between a handheld unit and a desktop or laptop unit is the types of peripheral devices available (full-sized keyboard and mouse versus touch screen, 20-in. monitor versus 2-in. screen), the amount of memory and hard disk storage space, and whether external storage is available such as flash drives via USB ports or optical disks via an optical disk drive.

Computers

We will study what makes up a computer in more detail in the next chapter. For now, we will look at the computer in more general terms. A computer is an electronic, programmable device. To run a program, the device needs a processor [Central Processing Unit (CPU)], memory to store the program and data, input and output capabilities, and possibly long-term storage and network capabilities (these last two are optional). Based on this definition, computers not only encompass desktop and laptop units, servers, mainframe computers, and supercomputers, but also netbooks, cell phones, computer game consoles, mp3 players, and book readers (e.g., Kindles). In the latter two cases, the devices are special-purpose—they run only a few select programs. The notion of the historical computer is gone. Today, we live with computers everywhere.

Figure 1.1 illustrates some of the range in computers. Desktop units with large monitors and system units are common as are laptop computers today with large monitors. Even more popular are handheld devices including personal digital assistants (PDAs), cell phones, and e-book readers. Monitors are flat screens. We no longer expect to find

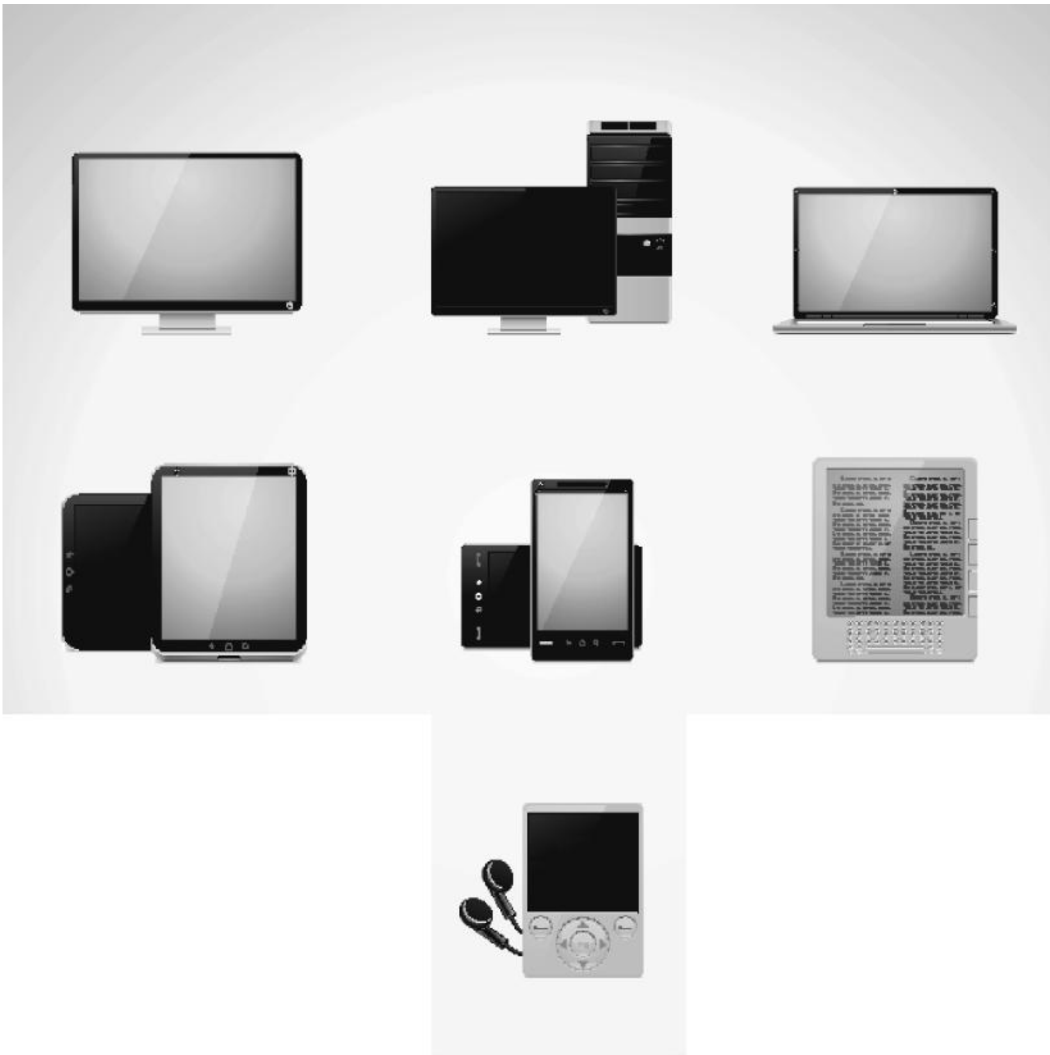

FIGURE 1.1 Types of computers. (Adapted from Shutterstock/tele52.)

bulky monitors on our desktop computers. Even so, the system unit, which allows us to have numerous disk drive devices and other components, is bulky. We sacrifice some of the peripheral devices when we use laptop computers. We sacrifice a greater amount of accessibility when we move on to handheld devices. In the case of the PDA, laptop, and notebook, the chips and motherboard, and whatever other forms of storage, must be placed inside a very small area. For the PDA, there is probably just a wireless card to permit access to the cell phone network (and possibly wi-fi). For the laptop and notebook computers, there is probably a hard disk drive. The laptop will also probably have an optical disk drive.

The main component of a computer is the *processor*. The processor's role is to process—that is, it executes the programs we run on the computer. To run a program on a given computer, the program has to be *compiled* for that computer. Compilation is a language translation process that converts a program from a more readable form (say Python or Java) into a computer's machine language. A computer can only run programs that are written in that machine's language. We discuss this concept in more detail, and the specific role of the CPU, in Chapter 2. We examine programming languages later in the textbook.

Aside from the processor, computers need storage. There are two types of storage—short-term storage and long-term storage. Short-term storage is most commonly random access memory (RAM). Unfortunately, RAM can describe several different types of memories. Our modern computers typically have three forms of RAM, dynamic RAM (what we typically call main memory), static RAM (cache memory and registers), and ROM (read-only memory). We differentiate between these types of memory in Chapter 2. For now, just consider all three to be "memory".

Main memory (dynamic RAM) is composed of chips. Dynamic RAM offers fast access and often large storage capacity. However, some handheld devices do not have room for much dynamic RAM storage. Instead, they use flash memory storage, which is more limited in capacity. Long-term storage most commonly uses hard disk drives but can also comprise optical disk, flash memory, and magnetic tape. Long-term storage is far greater in capacity than the short-term storage of main memory, and because additional storage space can always be purchased, we might view long-term storage capacity as unlimited. Typical desktop and laptop computer short-term storage is in the 1–8 GB range. 1 GB means 1 gigabyte, which is roughly 1 billion bytes. Think of a byte as 1 character (letter) such that 1 GB will store 1 billion characters. A typical book is probably on the order of 250,000 to 1 million characters. 1 GB would store at least 1000 books (without pictures). Hard disks can now store 1 TB, or 1 terabyte, which is 1 trillion bytes. Obviously, long-term storage is far greater in capacity than short-term storage. Some common storage sizes are shown in Table 1.4. We will study storage sizes in more detail in Chapters 2 and 3.

TABLE 1.4 Storage Sizes

Size	Meaning	Example
1 bit	A single 0 or 1	Smallest unit of storage, might store 1 black-and-white pixel or 1 true/false value, usually we have to combine many bits to create anything meaningful
1 byte (1B)	8 bits	We might store a number from 0 to 255 or –128 to 127, or a single character (letter of the alphabet, digit, punctuation mark)
1 word	32 or 64 bits	One piece of data such as a number or a program instruction
1 KB	1024 bytes	We might store a block of memory in this size
1 MB	~1 million bytes	A small image or a large text file, an mp3 file of a song might take between 3 and 10 MB, a 50-min TV show highly compressed might take 350 MB
1 GB	~1 billion bytes	A library of songs or images, dozens of books, a DVD requires several gigabytes of storage (4–8 GB)
1 TB	~1 trillion bytes	A library of movies

Do we need both forms of storage? It depends on the type of device and your intended usage, but in general, yes we need them both. Why? To run a program, we need to load that program into a storage space that responds quickly. Long-term storage is far slower than RAM, so unless you are willing to have a very slow computer, you need short-term storage. On the other hand, short-term storage is far more limited in capacity and the programs we run tend to be very large. We also often have very large data files (music files, movies, etc.) such that we cannot rely solely on short-term storage. Handheld devices offer a compromise—they often use flash memory instead of RAM, which results in a slower access time when compared to desktop/laptop computers, and they have a limited long-term storage space (if any), requiring that the user move files between the handheld devices and a permanent storage space (say on a desktop computer) fairly often.

Aside from the difference in speed and storage capacity between memory and long-term storage, another differentiating factor is their volatility. The term *volatile*, when describing memory, indicates whether the type of memory can retain its contents when the power supply has been shut off. Main memory (DRAM) and cache/register memory (SRAM) are volatile forms of memory. Once you turn the power off, the contents are lost. This is why, when you turn on your computer, memory is initially empty and the device must go through a "boot" process. Nonvolatile memories include ROM, flash drives, hard disk, and optical disk. The nonvolatile memory retains its contents indefinitely. In the case of ROM, the contents are never lost. In the case of flash drives and disks, the contents are retained until you decide to erase them.

Computers also require *peripheral* devices (although require is not the right word; perhaps we should say that for the user's convenience, we add peripheral devices). The word peripheral means "on the outskirts" but in the case of a computer, we usually refer to peripherals as devices that are outside the computer, or more specifically, outside of the system unit. The system unit is the box that contains the motherboard (which houses the CPU and memory) and the disk drive units. The peripherals are devices that are either too large to fit inside the system unit, or devices that must be accessible by the human users (Figure 1.2). These devices are our input and output devices—keyboard, mouse, track point, track ball or joystick, monitor, printer, speakers, pen and tablet (writing area) or light pen, etc. Without these input and output devices (known as I/O devices collectively), humans would not be able to interact with the computer. If all input data comes from a disk file and all output data similarly will be stored to disk file, there may be no need for the computer to interact with the human. But the human will eventually want to know what the program did.

Among the peripheral devices are the communication device(s). A communication device is one that lets a computer communicate with other computers. These devices are typically MODEMs, which can either require connection to a telephone line (or perhaps a cable TV coaxial line) or be wireless. Nearly all laptop computers today come with wireless MODEMs, whereas desktop units may come with a wired or wireless MODEM. However, in cases where computers are connected to a local area network (LAN), the computer requires a network connection instead of or in addition to the MODEM. The LAN connection is by means of a network card, often an Ethernet card. For high-speed networks,

FIGURE 1.2 Computer peripherals. (Courtesy of Shutterstock/Nevena.)

the network card offers a much higher bandwidth (transmission rate) than a MODEM. We will study wired and wireless MODEMs and network cards later in the textbook when we look at computer networks.

Let us now summarize our computer. A computer is in essence a collection of different devices, each of which performs a different type of task. The typical computer will comprise the following:

1. System unit, which houses
   a. The motherboard, which contains
      i. The CPU
      ii. A cooling unit for the CPU
      iii. Possibly extra processors (for instance, for graphics)
      iv. Memory chips for RAM, ROM
      v. Connectors for peripherals (sometimes known as ports)

vi. Expansion slots for other peripheral device cards

vii. The ROM BIOS for booting and basic input and output instructions

viii. Power supply connector

b. Disk drives

c. Fan units

d. Power supply

2. A monitor and keyboard

3. Typically some form of pointing device (mouse, track point, track ball)

4. Speakers (optional)

5. MODEM or network card (these are typically located inside the system unit, plugged into one of the expansion slots)

6. Printer (optional)

7. External storage devices such as external hard disk and tape drive

Chapter 2 has pictures to illustrate many of the above components.

Now we have defined a computer. But the computer is only a part of the story. Without software, the computer would have nothing to do. And without people, the computer would not know what program to run, nor on what data. So, our computer system includes these components.

Software

What is the point of a computer? To run programs. Without programs, the computer has nothing to do. A program, also known as *software* (to differentiate it from the physical components of the computer, the *hardware*), is a list of instructions that detail to the computer what to do. These instructions are written in a programming language, such as Java or Python. Programming language instructions must be very descriptive. For instance, if you want the computer to input two numbers from the user and output which one is larger, you could not just say “input two numbers and output the larger of the two.” Instead, you must describe the actions to take place as an algorithm, broken into step-by-step instructions. The instructions must be written in a programming language. For instance, the problem described in this paragraph might be broken into four steps:

Input number1

Input number2

Compare the two numbers and if the first is greater than the second, output number1

Otherwise output number2

In a language like C, this would look like this:

```
scanf("%d", &number1);
scanf("%d", &number2);
if(number1 > number2) printf("%d is greater", number1);
else printf("%d is greater", number2);
```

The scanf instruction inputs a value, the printf instruction, outputs a value or message. The if instruction is used to compare two values and make a decision. Some of the syntax in C is peculiar, for instance the & before "number1" and "number2" in the scanf statements, the use of the semicolon to end instructions, and the use of %d. Every language will have its own syntax and in many cases, the syntax can appear very odd to someone who is not a programmer or has not learned that language.

Programs are not just a list of executable statements. Programs also require various definitions. These might include variable declarations, functions or methods, and classes. In C, for instance, we would have to define number1 and number2 as being variables to be used in the above code. In this example, they would be declared as integer numbers.

There are many forms of software, but we generally divide them into two categories: system software (the operating system) and application software (programs that we run to accomplish our tasks such as a word processor, an Internet browser or a computer game). Usually, our software is written by professionals—software engineers. However, once you learn to program, you can write your own software if you desire. As an IT student, you will learn to write short pieces of code, scripts. Scripts can be used to support either the operating system or an application. For instance, you might write a Bash shell script to support an activity such as automatically creating user accounts for a new group of users. Or you might write a server-side script in Perl to test a URL for security threats in a web server.

Users

Without the human, the computer would not have anything to do. It is the user who initiates the processes on the computer. "Do this now, do that later." We may want to interact with the programs while they run. This interactivity is done through the I/O devices. Today, we are so used to interactivity that we probably cannot imagine using computers without it. But in earlier days (1940s–1970s), most—if not all—processing was done without human interaction at all. The user specified the program, the source of input, the location of output, and sent the program off to run. The user would see the results once the computer ran the program, which might have been immediately, or many hours later!

Users have progressed over time, just as the technology has progressed. The earliest computer users were the engineers who built and programmed them. Computers were so complicated and expensive that no one else would have access. As computer costs permitted organizations to purchase them (for millions of dollars), computer users were those employees who had received specialized training to use them. Things began to change with the advent of personal computers, first released in the 1970s. But it was not until windowing operating systems came about that computer users could learn to use the computers with little to no training. And so today, it is common that anyone and everyone can use a computer. In fact, computers are so commonplace that people may not realize that they are using a computer when they program their GPS or run an application on their smart phone.

Our View Today

Computers used to be easily identifiable. They were monstrously expensive devices that would weigh tons, filled up a room or more, required clean room environments and special air conditioning. People would not actually touch the computer; they would interface with the computer through terminals and networks. With personal computers, computers for individuals became affordable and many people began to have computers in their own homes. Telecommunication, over LANs or over telephone networks, allowed people to connect their computers together, to communicate to each other and share e-mail messages, files, programs, etc. The Internet, which was first turned on in its earliest form in 1969, became commercially available to home computer users in the mid-1990s. Early in this period, people connected to the Internet via slow MODEM access over their telephones. But over the past 15 years, telecommunications has changed completely. Now, we have wireless access, high-speed Internet connections, cell phones, and more.

Today, computers are not easily identifiable. They are no longer limited to mainframe computers or desktop units. You can have a network computer or a laptop, a notebook computer, a tablet computer, a handheld computer. We even have devices smaller than handheld units that use processors and memory. And our connectivity has changed equally. Your access to telecommunications is no longer limited by the telephone port in your home. With wireless, you can gain access anywhere in your household or anywhere in a remote location that has hot spots. Want a coffee break? No problem, go to Starbucks and you can still access the Internet through your laptop. Or, taking a drive? You can still access the Internet over your cell phone (as long as you are in reasonable proximity to a cell phone tower). We are a world united through nearly instantaneous communication no matter where we are. And we are a world of billions upon billions of processors. We used to count computers by the millions, but today, there are tens of billions of processors and most of these can communicate with each other.

This gentle introduction to IT will serve as our starting point in this text. Over the chapters to come, we will study many IT-related concepts. We first look at computer components, gain an understanding of what they do, how they work, and how we connect them to a computer. We also study a related topic, binary numbers and how we use binary. These next two chapters on computer organization and binary are often material covered in computer science curricula. They are included here so that, as an IT person, you understand more than what a processor and a motherboard are when it comes to the computer hardware. By having a firm foundation of what the computer components do and how they work, you should be able to understand the necessity of when to increase RAM, or how to evaluate a processor. The inclusion of binary in this text is largely to support concepts found in computer networks.

The focus shifts to system software, that is, operating systems. We examine two of the most common operating system platforms: Windows (Windows 7) and Unix (Red Hat

Linux). We will compare and contrast what they look like, how we use them, and how we configure them. Operating system topics include file systems, users, accounts and permissions, processes and process management, and services. We also examine two Linux-specific topics: the Bash shell and the use of regular expressions in Linux.

The text examines several different histories. The evolution of computer hardware, the evolution of operating systems, the evolution of computer programming, the history of both Linux and Windows, and the history of the Internet are all covered (although not in the same chapter). Although perhaps not necessary for an IT person, it does help set a backdrop to how technology has changed so that you will have an appreciation of the rapidity behind the changes in IT. Additionally, by understanding the past, it might help you understand where IT might lead.

The final collection of chapters covers other IT topics. Computer networks are considered from several different perspectives. The logical structure of a network, the physical nature of a network, the network protocols that proscribe how transmitted data are to be treated, and some of the more common network software are all examined in one chapter. Software management describes the types of software available and provides details for how to install software in a computer system, particularly in Linux with open source software. Another chapter concentrates on programming, offering examples of writing scripts in both the Linux shell and DOS. The penultimate chapter of the text covers the information side of IT. In this chapter, we examine such ideas as information management and information assurance and security. A final chapter wraps up the text by considering careers in IT and various topics related to IT professionals.

FURTHER READING

There are a number of websites that provide information on IT careers, some of which are listed below.

- http://www.wetfeet.com/careers-industries/careers/information-technology
- http://www.cio.com/article/101314/The_Hottest_Jobs_In_Information_Technology
- http://www.careeroverview.com/technology-careers.html
- http://www.techcareers.com/
- http://information-technology.careerbuilder.com/

The best source for IT education can be found through the special interest group on IT education (SIGITE) at http://www.sigite.org/it-model-curriculum.

General introductions to computer hardware, software, and users can be found in any number of computer literacy texts such as these:

- Beekman, G. and Beekman, B. *Tomorrow's Technology and You*. Upper Saddle River, NJ: Prentice Hall, 2008.

- Fuller, F. and Larson, B. *Computers: Understanding Technology*. St. Paul, MN: ECM Paradigm Publishing, 2010.
- Meyer, M., Baber, R., and Pfaffenberger, B. *Computers in Your Future*. Upper Saddle River, NJ: Prentice Hall, 2007.
- Laberta, C. *Computers Are Your Future*. Upper Saddle River, NJ: Prentice Hall, 2011.
- Williams, B. and Sawyer, S. *Using Information Technology*. New York: McGraw-Hill, 2010.
- Snyder, L. *Fluency with Information Technology: Skills, Concepts and Capabilities*. Upper Saddle River, NJ: Prentice Hall, 2010.

However, as someone who wishes to make a career of IT, you would be better served with more detailed material. Such texts will be listed in later chapters as we cover material in greater depth. See the further readings in Chapter 2 for more information on computer hardware, Chapter 4 for more information on operating systems, Chapter 14 for more information on programming, and Chapter 16 for more information on IT careers.

REVIEW TERMS

The following terms were introduced in this chapter:

Administrator	Peripheral
Computer	Processor
Hardware	Network administrator
Help desk	Software
Information Technology	Storage capacity
IT specialist	System administrator
MODEM	User

REVIEW QUESTIONS

1. What are the skills expected of an IT specialist?
2. What does administration mean in reference to IT?
3. What does training mean in reference to IT?
4. How does the study of IT differ from a 2-year technical degree in computers?
5. To what extent should an IT specialist be able to write computer programs?

6. What is a system administrator? What is a network administrator? How do the two jobs differ?
7. Define a computer.
8. What is the IPOS cycle?
9. Should a cell phone be considered a computer?
10. How does a computer system differ from a computer?
11. How do short-term and long-term storage differ?
12. What is software? What are the two general forms of software?

DISCUSSION QUESTIONS

1. As a student of IT, what brought about your interests in studying IT? Having read this chapter, are you as interested in IT as you were before, more interested or less interested?
2. Organize the IT skills listed in Table 1.2 in order of most important to least important for a system administrator. Defend your listing based on the types of tasks that a system administrator will be required to undertake.
3. Table 1.2 did not include "soft skills" such as the ability to communicate with others, the ability to work in groups, and the ability to manage projects. Are these types of skills taught or are they learned in other ways? Should a 4-year IT program include courses that cover such skills?
4. What are the differences between computer information technology and computer science? Should a program in computer science include computer information technology courses, or should they be separate programs?
5. How does a 4-year IT degree differ from a 2-year IT degree or a degree earned at an IT technical school?
6. Compare computers of today to those that existed in the 1950s.
7. In your lifetime, what changes have you seen in computers and other information technology (particularly handheld devices)? What changes do you expect to see in the next 10–15 years?
8. Many people are surprised to learn that smart phones should be considered computers. In what ways are smart phones similar to desktop and laptop computers? In what ways are they different? Should ordinary cell phones be considered computers?