Chapter 0 Introduction

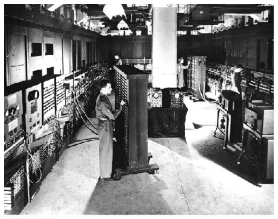
0.1 A Brief History of Computers

- What is a **computer**?
  - A mechanical or electronic device
  - Stores, retrieves, manipulates large amounts of information at high speed, with great accuracy
  - Does not need human intervention
  - Carries out instructions from a **program**

The Pioneers

- Mid-1800's: Charles Babbage built the **Analytical Engine**
  - made from axles and gears that could store and process 40 digit numbers
  - assisted by Ada Byron who has a major programming language named after her (RAPTOR built on Ada)
- 1940: Howard Aitken at Harvard, with John Atanasoff and Clifford Berry at Iowa State U. created **Mark I**, an electronic computer.
  - could not act on intermediate results.
- 1945: John Mauchly and J. Presper Eckert at U. Pennsylvania built the **ENIAC** (Electronic Numerical Integrator and Calculator)
  - weighed 33 tons, 17,000 vacuum tubes
  - performed up to 5000 additions per second

ENIAC: the computer of the 1940's!



Source: U.S. Army

Early Computers: 1940's – 1950's

1945 – 1950's: First generation computers

- used **vacuum tubes** to do internal switching needed for computations
- 1955: about 300 computers in the world based on vacuum tubes
- Late 1950s: invention of the **transistor** was one of most important inventions of 20th Century
  - computers based on the transistor are the first solid-state computers
  - need climate-controlled environment

The Personal Computer

- 1970s: The personal computer becomes available with invention of the **microchip**
- 1974: The microchip, along with the invention of the **microprocessor** led to creation of first personal computer
- Bill Gates and Paul Allen founded Microsoft Corporation
- Stephen Wozniak and Steven Jobs founded Apple Computer, Inc.

Computers Today

- **Supercomputers** are very powerful and specialized and are used for massive computing problems by big corporations and government departments
- **Mainframes** are in use at large corporations
- **Desktop computers** and **Laptops**:
  - **PC**s: computers that use the Microsoft Windows operating system
  - **Mac**s compete with PCs in the personal computer market.
- **Smart phones**: the power of a computer combined with the lure of a cell phone
- **Tablets**: all the features most users want in a computer combined with portability

The Internet

Internet – a world-wide collection of networks
- **network**: 2 or more linked computers
- roots of the Internet: 1960's, U.S. Defense Department project

Email: electronic mail

WWW: **World Wide Web**, originated 1989
- A vast collection of linked documents (**Web pages**)

Web2.0 – Social Networking
- consists of Web applications that facilitate information sharing, user-centered design, and collaboration

0.2 Computer Basics

Components of a computer
- **Central Processing Unit** (CPU)
- **Internal memory**
  - RAM (Random Access Memory)
  - ROM (Read Only Memory)
- **Mass storage devices**
  - Magnetic, optical, and solid-state and the Cloud
- The **system unit** houses the CPU, internal memory, and most mass storage
- **Input devices** such as keyboard and mouse
- **Output devices** such as monitor and printer

Central Processing Unit (CPU)

CPU is often called the **brain of the computer**
- Receives program instructions
- Performs arithmetic and logical operations
- Controls other computer components

Consists of millions of transistors on a single microchip that plug into the **motherboard**

Speed of CPU is measured in **gigahertz** (GHz)

Internal Memory (RAM and ROM)

ROM: read-only memory
- Contains instructions used by computer during startup
- Cannot be altered by computer user

RAM: random-access memory
- Is a "scratch pad" for user as he or she works
- Can be read from and written to
- RAM is measured in **megabytes** (MB) or **gigabytes** (GB)

Basic Units of Memory

- 1 **bit** (0 or 1)
- 1 **byte** normally consists of 8 bits
  - Is the storage for one character
- 2^{10} bytes =1024 bytes = 1 **kilobyte** (**KB**)
- 1024 KB = 1 **megabyte** (**MB**)
- 1024 MB = 1 **gigabyte** (**GB**)
- Many people approximate in steps of 1000, not 1024
  - Example: a 20KB file actually has 20,480 bytes, not 20,000 bytes

Mass Storage Devices

Magnetic storage
- Hard disks are always internal but external hard disks may be added as suppllemental storage

Optical storage
- CDs and DVDs

Solid-state storage
- Flash drives plug into a USB port

Cloud Computing
- Delivers computer services over the Internet with a host that provides service:
  - Infrastructure (hardware, servers, networking)
  - Platform (rent hardware, operating systems, storage, networking capacity)
  - Software (use software applications for a fee)

Input and Output Devices

Input
- Keyboard
- Mouse
- digital pen
- Modem or wireless Internet connection
- touch screen
- joy sticks
- Voice command
- …and more

Output
- Monitor
- Printer
- Speakers
- modem or wireless Internet connection
- …and more

0.3 Software and Programming Languages

Application Software
- enhances productivity
- solves problems
- supplies information
- provides entertainment
- examples: word processors, database managers, spreadsheets, photo editors, browsers

System Software: The Operating System
- controls and maintains hardware
- communicates with user
- manages and communicates with applications
- examples: Windows, DOS, Linux, UNIX

Programming and Scripting Languages

Machine Languages
- Consists only of sequences of 0s and 1s
- example:

0110110111110111 0000000100000000
0000000100000000

Assembly Languages
- symbolic representation of machine language
- example: ADD A,B

High-level Languages
- contains English words and phrases and algebraic expressions
- examples of high level languages:

C++	Objective C
COBOL	Java
JavaScript	Visual Basic

Scripting Languages
- Interpreted, not compiles
- Client-side (such as JavaScript)
- Server-side (such as PHP or ASP)

Writing Programs

To write a program in a high-level language, you need:
- appropriate **software**
- a **text editor** to type and edit program statements
- a **debugger** to help find errors in program code
- a **compiler** or **interpreter** to translate the program into machine language

Programming Logic

- All programming languages use basic programming logic.
- If you understand this logic, it will be much easier to learn any specific language.