Overview

Hui Chen

Department of Computer & Information Science

Brooklyn College

Objectives

- To understand computer basics, programs, and operating systems
- To understand the meaning of Java language specification, API, JDK, and IDE
- To write a simple Java program
- To display output on the console
- To explain the basic syntax of a Java program
- To create, compile, and run Java programs

Outline

- What is a computer?
- What is a program?
- What is a programing language?
- How do we let a computer run a program?
- What is an operating system?
- Why Java?
- How do we write a simple Java program?

What is a computer?

What is a computer?

- An electronic device that stores and process data, including both
  - Hardware
  - Software (programs and data)

Hardware Components of Computers

- Central Processing Unit (CPU)
- Memory and Storage
- Input and Output Devices
- Communication Devices

Central Processing Unit (CPU)

- Built on small silicon semiconductor chips (contains transistors)
- CPU core
  - Control unit (CU)
  - Arithmetic/logic unit (ALU)
  - Multi-core CPUs
- Clock speed
- Cache

Memory and Storage

- Bits and Bytes -- How Data are Stored?

How Data are Stored?

- Representation
  - Binary representation
  - Bits and Bytes
  - Units of measurement
- Addressing

Bits and Bytes

- A computer has a lot of tiny electronic switches (called transistors), each switch exists in two states:
    - On (representing 1)
    - Off (representing 0)
- They are digits in the binary number system, we call them “bits”
- The minimum addressable storage unit is a “byte”, composed of 8 bits

Measurement Storage Capacity and Data

- Byte
- Kilobyte (KB)
- Megabyte (MB)
- Gigabyte (GB)
- Terabyte (TB)
- …

Memory and Address

- A computer’s memory is organized as an ordered sequence of bytes (for storing programs and data)
  - Each byte has a unique address, that is used to locate the byte

Memory Hierarchy

- Roles
  - Main memory (often memory)
  - Secondary storage (often storage)
  - Tertiary storage (backup)
- Characteristics
  - Volatile vs non-volatile
  - Access latency (speed)
  - Capacity (size)
  - Reliability (e.g., MTBF)

Input and Output Devices

- Input devices
    - Mouse, keyboard, touch pad, joy sticker, …
- Output devices
    - Monitor
        - Resolution
        - Dot pitch
    - Printer

Communication Devices

- Examples
  - Bluetooth
  - Ethernet
  - Wireless LAN (e.g., Wi-Fi)

Questions?

Computer Software

- (Computer) hardware vs. (computer) software
- Programs are software
- Computer programs
  - Instructions to a computer to tell a computer what to do, written in a programming language
- Programming language
  - A language we use to communicate with a computer

Programming languages

- Machine language
    - A computer’s native language, are in the form of binary code, e.g.,
    - Adding two numbers may be:
        - 1011010111010101
- Assembly language
    - Mnemonic form of a machine language
    - Adding two numbers becomes more readable, also easier to write, like
        - add 2, 3, result
- High-level language

Examples of High-Level Languages

How do we let a computer run a program?

- How do we let a computer run a program written in a high-level (or assembly) language?
  - Source code, interpreter, and compiler
  - Interpreting
  - Compilation

What is an operating system?

- Examples of operating systems?
- What does an operating system do?

Questions?

Why Java?

- Java is a general purpose high-level programming language? But why?
- JRE vs JDK

Questions?

Begin to writing simple Java programs

- Programming environment
- Write a simple Java program
- Compile and test

Writing a simple Java program

- Preferred development environment
  - Git bash + Atom editor + JDK 1.8 or newer
- (Optional) Using IDEs
  - IntelliJ IDEA
  - Eclipse
  - NetBeans
  - …
- https://www.sci.brooklyn.cuny.edu/~goetz/java/

Let’s write the “Hello, World” program

- Class name
- Main method
- Statements
- Statement terminator
- Reserved words
- Comments
- Blocks

Compile and Test

- Compile the program
  - javac HelloWorld.java
  - Is there any error?
  - What is the result of compilation (with or without errors)?
    - .java file (source code) and .class file (bytecode)
- Test versus run the program?
  - We run the program to test it
  - java HelloWorld

Questions?

What if there is an error?

- What kind of error?
  - Compilation errors
  - Runtime errors
  - Logic errors
- How do we deal with errors?
- Let’s write a few more programs

Given two numbers, compute the division

- Write the program
- What if the denominator is 0?
    - An example of runtime error – a situation the program cannot handle

Given a temperature in Celsius, convert it to Fahrenheit

- Write the program
- Does the program produce an expected result?
    - An example of logic error – the situation that the program does not perform the way it was intended to.

Questions?

Lab Exercises

- Exercise A
- Exercise B
- CodeLab Registration

Lab Exercise A

- Preparing programming environment
  - Download and set up Git, Atom, JDK
  - Optionally, using an IDE
- To test that you have a working Java programming environment, complete Exercise B

Lab Exercise B

- Write, compile, and run the “Hello, World” java program
    1. Create/Revise the program
    2. Compile the program
    3. Run the program
- Introduce errors on purpose, e.g.,
    - remove “;”
    - remove “(“
    - remove “}”
    - change “main” to “primary”)
    - …