Chapter 1 Introduction to Computers, Programs, and Java

What is a Computer?
- A machine that manipulates data according to a list of instructions
- Consists of hardware and software
- Many physical forms

A Bit of History
- 1801 first programmable device
- 1930s and 1940s digital computers
What is a Computer?

- Computer components: CPU, memory, hard disk, floppy disk, monitor, printer, and communication devices.

CPU

- Stands for central processing unit
- Retrieves instructions from memory and executes them
- Speed measured in GHz (gigahertz) (1GHz=1 billion of pulses per second)
- Modern CPU can have few cores

Memory (RAM)

- Stores data and program instructions for CPU
- Memory unit: byte (a sequence of 8 bits)
  - What is a bit? It's a binary digit taking value 0 or 1
  - Memory is a volatile storage
- Programs and data need to be loaded to main memory before execution by CPU
How Data is Stored in Memory?

- Everything is coded as a series of bits
- Why computers use only 0/1?
- Programmers need not to be concerned about the encoding and decoding of data
- Performed automatically

<table>
<thead>
<tr>
<th>Memory Address</th>
<th>Memory Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>101111</td>
</tr>
<tr>
<td>2001</td>
<td>011010</td>
</tr>
<tr>
<td>2002</td>
<td>111001</td>
</tr>
<tr>
<td>2003</td>
<td>010110</td>
</tr>
</tbody>
</table>

Storage Devices

- Reminder: main memory is volatile
- Programs and data are permanently stored on storage devices or secondary memory
- Storage device types:
  - Disk drive
  - CD...

Input and Output Devices

- I/O is the means by which a computer receives information from the outside world and sends result back.
- Output devices: monitor, printer
- Input devices: keyboard, mouse, joystick …
Getting rid of your keyboard and mouse?

The Mac Wheel

Computer Programs

- Computer program is a set of instructions
- Operating systems: core software of a computer
  - Windows (XP, Vista), Unix, Linux, MacOS, …
- Application software: any software other than operating systems
  - Web browsers, word processors, games, …

Operating Systems (OS)

- A program that manages and controls a computer’s activities
- Any other program (internet browser, word processor) is managed by OS
- The programs you write are also managed by OS

What are the tasks of OS?

1. Controlling and monitoring system activities
2. Allocating and assigning system resources
3. Scheduling operations
Program Languages

- Instructions are specified using a computer programming language
- Three types of programming languages:
  - Machine language
  - Assembly language
  - High-level language

Machine Language

- Set of primitive instructions built into every computer
- Instructions are in the form of binary code
- Why don’t we use those languages?

Geek 1
I wrote this awesome program: 10001110

Geek 2
No, no, no. It should be: 10000110

Just a human being

Assembly Language

- Developed to make programming easy
- The computer cannot understand assembly language
- Need to convert assembly code to machine code

```
ADDF R1, R2, R3
```
High-Level Language

- English-like and easy to learn and program
- Can be understood by just looking at the code

area = 5 * 5 * 3.1415;

- Required steps to produce byte code?

Compiling Source Code

- Source code
  - A program written in a high-level language
- Compiler
  - Used to translate the source program into a machine language program called an object program
- Linker
  - Used to link your object program with supporting libraries

Some Popular High-Level Languages

- COBOL (COmmon Business Oriented Language)
- FORTRAN (FORmula TRANslator)
- BASIC (Beginner All-purpose Symbolic Instructional Code)
- Pascal (named for Blaise Pascal)
- Ada (named for Ada Lovelace)
- C (whose developer designed B first)
- Visual Basic (Basic-like visual language developed by Microsoft)
- Delphi (Pascal-like visual language developed by Borland)
- C++ (an object-oriented language, based on C)
- C# (a Java-like language developed by Microsoft)
- Java (We will use this one!!!)
Java's History

- James Gosling and Sun Microsystems
- Java presented on May 20, 1995 during Sun World
- HotJava
  - The first Java-enabled Web browser
- Why the name Java?

Silk Lyric Pepper NetProse
Neon Java DNA WebDancer
WebSpinner WRL (WebRunner Languages)

Why Java?

- Applications for:
  - Internet services (Java Applets, Java Web Applications)
  - Desktop computers
  - Hand-held devices
- Well-established and popular language
  - Langpop.com

Characteristics of Java

- Java Is Object-Oriented
- Java Is Distributed
- Java Is Robust
- Java Is Architecture-Neutral
- Java Is Portable
- Java Is Secure
- Java Is Multithreaded
Characteristics of Java

- Designed to be Object-Oriented
- Object-Oriented programming provides great flexibility, modularity, clarity, and reusability

- Java Is Object-Oriented
- Java Is Distributed
- Java Is Robust
- Java Is Architecture-Neutral
- Java Is Portable
- Java Is Secure
- Java Is Multithreaded

Distributed computing involves several computers working together on a network
Designed to make distributed computing easy

Compilers can detect many problems that would first show up at execution time in other languages
A runtime exception-handling feature
Characteristics of Java

- Java Is Object-Oriented
- Java Is Distributed
- Java Is Robust
- Java Is Architecture-Neutral
- Java Is Portable
- Java Is Secure
- Java Is Multithreaded

- Write once, run anywhere

- Java Is Object-Oriented
- Java Is Distributed
- Java Is Robust
- Java Is Architecture-Neutral
- Java Is Portable
- Java Is Secure
- Java Is Multithreaded

- Implements several security mechanisms to protect your system against harm caused by stray programs

- What is multithreading?
  - Multithread programming is smoothly integrated
Review Questions – what is a computer

1. ________ is the physical aspect of the computer that can be seen.
   A. Hardware  
   B. Software  
   C. Operating system  
   D. Application program

2. ________ is the brain of a computer.
   A. Hardware  
   B. CPU  
   C. Memory  
   D. Disk

3. Which of the following are input and output devices?
   A. Keyboard and mouse  
   B. CPU  
   C. flash stick  
   D. Software

Review questions – computer programs

1. ________ are instructions to the computer.
   A. Hardware  
   B. Software  
   C. Mouse  
   D. Keyboards

2. Computer can execute the code in ________.
   A. machine language  
   B. assembly language  
   C. high-level language  
   D. none of the above

Java standard

- Language specification
  - Syntax and semantics of Java language
- API (Application Program Interface)
  - Predefined classes for developing java programs
JDK Editions

- **Java Standard Edition (J2SE)**
  - J2SE can be used to develop client-side standalone applications or applets.

- **Java Enterprise Edition (J2EE)**
  - J2EE can be used to develop server-side applications such as Java servlets and Java ServerPages.

- **Java Micro Edition (J2ME)**
  - J2ME can be used to develop applications for mobile devices such as cell phones.

We will be using J2SE

JDK Versions

- JDK 1.02 (1995)
- JDK 1.1 (1996)
- JDK 1.2 (1998)
- JDK 1.3 (2000)
- JDK 1.4 (2002)
- JDK 1.5 (2004) a. k. a. JDK 5 or Java 5
- JDK 1.6 (2006) a. k. a. JDK 6 or Java 6

Creating, Compiling, and Running Programs

Source code (developed by the programmer)

- Process the source code to produce an object file.

- The object file is read by the Java Virtual Machine (JVM) to run the program.

- Output is generated and displayed on the screen or written to a file.

 bytecode (generated by the compiler for JVM to read and interpret, not for you to understand)

- Source code (developed by the programmer)

- Process the source code to produce an object file.

- The object file is read by the Java Virtual Machine (JVM) to run the program.

- Output is generated and displayed on the screen or written to a file.

- Result

```java
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Method Welcome()

- aload_0

- getstatic #2 ...

- ldc #3 <String "Welcome to Java!">

- invokevirtual #4 ...

- return

- Result
Programming Environment in Unix

- Unix operating system
  - Command line interface (CLI) through shell window
  - GNOME desktop environment
- Unix editors
  - vi or Vim
  - gedit
- Java compiler
  - javac included in JDK
- Java Virtual Machine (JVM)
  - java

Understand Files and Folders

- Java source code are saved as .java files
- Files: a collection of items of information that are kept together
- Files have names, and the rules for legal names differ from one system to another
- Files are stored in folders or directories; these file containers can be nested

Integrated Development Environment (IDE)

- IDE is a software application that provides comprehensive facilities for software development
  - Source code editor
  - Compiler and/or interpreter
  - Debugger
- Popular IDEs
  - Eclipse Open Source by IBM
  - NetBeans Open Source by Sun
  - Borland JBuilder 2007
Programming environment in Windows

- See Supplement I.B for installing and configuring JDK
- See Supplement I.C for compiling and running Java from the command window for details

www.cs.armstrong.edu/liang/intro7e

A Simple Java Program

Listing 1.1

```java
// This program prints Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Demo

- Create a source program with an editor
  - gedit Welcome.java
- Save the program to hard disk
- Compile the program with a compiler
  - javac Welcome.java
- Execute the program with java JVM
  - java Welcome
Creating and Editing Using gedit

To use gedit, type `gedit Welcome.java` from the shell.

```
//This program prints Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Trace a Program Execution

Enter main method

```
//This program prints Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Execute statement

```
//This program prints Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```
Trace a Program Execution

```java
//This program prints Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Anatomy of a Java Program

- White spaces
- Comments
- Package
- Reserved words
- Modifiers
- Statements
- Blocks
- Classes
- Methods
- The main method

Comments

- Three types of comments
  - Line comment
    `//This is an example of line comment`
  - Paragraph comment
    `/# This is an example of paragraph comment. It can have few lines...#/
  - Javadoc comment
    `/** This is a Javadoc comment. */`
Package

- Packages help to organize code
- Package is always declared at the beginning of the code

```java
package com.yourpackage;
import java.io.*;

// This class prints string Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        // The statement below prints the string
        System.out.println("Welcome to Java!");
    }
}
```

Reserved Words

- Also known as keywords
- Words that have a specific meaning to the compiler
- Cannot be used for other purposes in the program

```java
package com.yourpackage;
import java.io.*;

// This class prints string Welcome to Java!
public class Welcome {
    public static void main(String[] args) {
        // The statement below prints the string
        System.out.println("Welcome to Java!");
    }
}
```

Modifiers

- Certain reserved words are called modifiers
- They specify the properties of the data, methods, and classes
- Examples of modifiers are public and static, private, final, abstract, and protected

```java
package com.yourpackage;
import java.io.*;

// This class prints string Welcome to Java!
public static class Welcome {
    public static void main(String[] args) {
        // The statement below prints the string
        System.out.println("Welcome to Java!");
    }
}
```
Statements

- A statement represents an action or a sequence of actions

```
package ej12;
import java.io.*;

/**
 * This class prints string Welcome to Java!
 */
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

- Other statement: area = r * r * 3.14

---

Blocks

- A pair of braces in a program forms a block that groups components of a program

```
public class Test {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

---

Classes

- The class is the essential Java construct
- A class is a template or blueprint for objects
- We will learn a lot about classes later in the course

```
package ej12;
import java.io.*;

/**
 * This class prints string Welcome to Java!
 */
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```
Methods

- Method is a collection of statements
- Methods perform a sequences of operations
- Used by invoking a statement with a string argument
- The arguments are enclosed within parentheses and separated by commas

```java
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Programming is fun!");
        System.out.println("Fundamentals first");
        System.out.println("Problem driven");
    }
}
```

main Method

- The main method provides the control of program flow
- The Java interpreter executes the application by invoking the main method

```java
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java");
    }
}
```

Another example

```java
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Programming is fun!");
        System.out.println("Fundamentals first");
        System.out.println("Problem driven");
    }
}
```
Class Exercise

- Write a Java program that displays a smiley face.

```
//////////
| 0 0 |
|    ^|
|    U |
|_____|
```

Displaying Text in a Message Dialog Box

- Java can be used for rapid application development.
- For instance, you can create message dialog boxes very fast.
- Simply use the `showMessageDialog` method in the `JOptionPane` class.
- `JOptionPane` is one of the many predefined classes in the Java system.

```
JOptionPane.showMessageDialog(null, "Welcome to Java!", "Display Message", JOptionPane.INFORMATION_MESSAGE);
```

The `showMessageDialog` Method
Summary

- Computer architecture and components
- Programs and Programming languages
- Operating System
- Java advantages
- Programming environment
- The first Java program

- Binary and decimal number system (readings)