



# Introduction to Java

Rajesh Palit, Ph.D.

Electrical and Computer Engineering

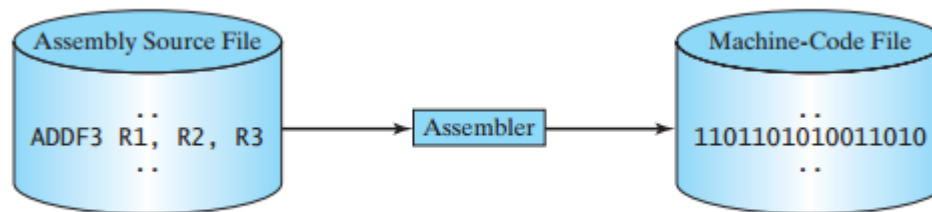
North South University

# Computer Programming

- A computer program is a set of instructions to the computer telling it what to do. Programming is the creation of a program executable by a computer and performs the required tasks
- Computers do not understand human languages, so we need to use computer languages in computer programs
- A computer's native language is a set of built-in primitive instructions. The instruction set is in the form of binary code and differs among different types of computers
- Programming in machine language is a tedious process. Moreover, the programs are highly difficult to read and modify
- To add two numbers, we might have to write an instruction in binary like this 1101101010011010

# Assembly Language

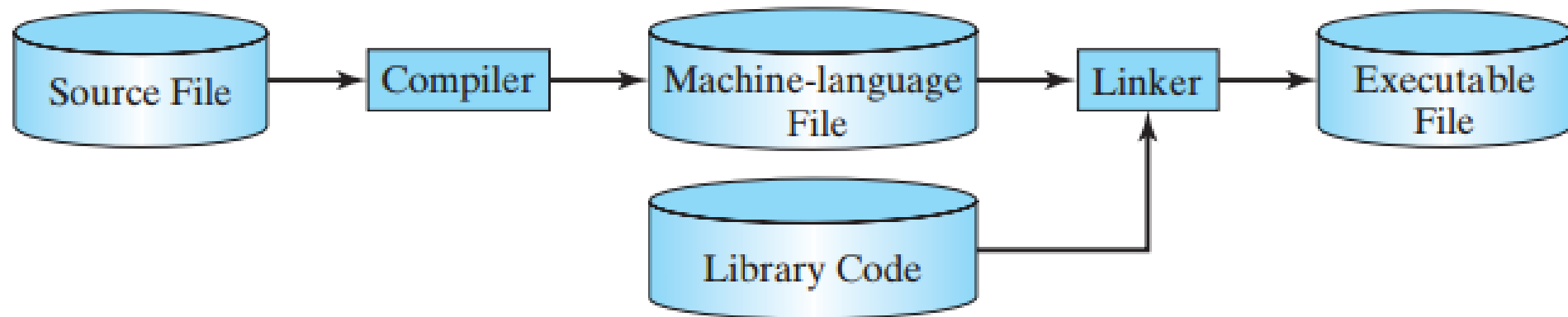
- Assembly language is a low-level programming language in which a mnemonic is used to represent each of the machine language instructions. For example, to add two numbers, we might write an instruction in assembly code like this: `ADDF3 R1, R2, R3`
- Since the computers cannot understand assembly language, a program called an assembler is used to convert assembly-language programs into machine code



Assembler translates assembly-language instructions to machine code.

# High Level Programming Languages

- As program size grows, assembly program becomes unmanageable
- The high-level languages are English-like and easy to learn and program. Here, for example, is a high-level language statement that computes the area of a circle with radius 5:       $\text{area} = 5 * 5 * 3.1415;$



# Prominent High Level Programming Languages

- ❑ COBOL (COmmon Business Oriented Language)
- ❑ FORTRAN (FORmula TRANslation)
- ❑ BASIC (Beginner's All-purpose Symbolic Instruction Code)
- ❑ Pascal (named for Blaise Pascal)
- ❑ Ada (named for Ada Lovelace)
- ❑ C (developed by the designer of B)
- ❑ Visual Basic (Basic-like visual language 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

# Why Java?

- ❑ The answer is that Java enables users to develop and deploy applications on the Internet for servers, desktop computers, and small hand-held devices
- ❑ The future of computing is being profoundly influenced by the Internet, and Java promises to remain a big part of that future. Java is the Internet programming language
- ❑ Java is a general purpose programming language as well as the Internet programming language
- ❑ Java is purely Object Oriented Programming Language

# History of Java

- James Gosling, Mike Sheridan, and Patrick Naughton initiated the Java language project in June 1991 in Sun Microsystems
- Java was originally designed for interactive television, but it was too advanced for the digital cable television industry at the time
- The language was initially called Oak after an oak tree that stood outside Gosling's office
- It went by the name Green later, and was later renamed Java, from Java coffee said to be consumed in large quantities by the language's creators
- Public in Sun World: May 20, 1995

# Java Development Kit (JDK) Versions

- JDK 1.02 (1995)
- JDK 1.1 (1996)
- JDK 1.2 (1998)
- JDK 1.3 (2000)
- JDK 1.4 (2002)
- J2SE 5 (2004), internal version 1.5 or Java 5
- J2SE 6 (2006), internal version 1.6 or Java 6
- J2SE 7 (2011)
- J2SE 8 (2014)
- Java 2 Platform, Standard Edition 12 (March 2019)

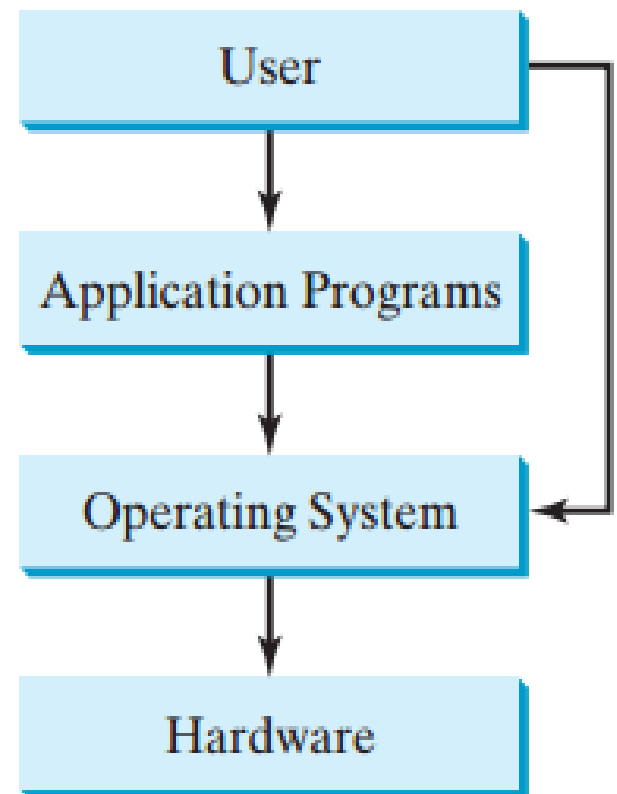


# JDK Editions

- 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.
- This book uses J2SE to introduce Java programming.

# Application Program and Operating Systems (OS)

- The operating system(OS) is the most important program that runs on a computer, which manages and controls a computer's activities
- OS abstracts the hardware from the users
- Users do not need to know the details of the HW
- Using application programs such as a web browser or a word processor, users utilize hardware components through OS



- Write a program
- Compile the program
- Hello.c => compiler => hello.exe
- Hello.exe => Windows 10

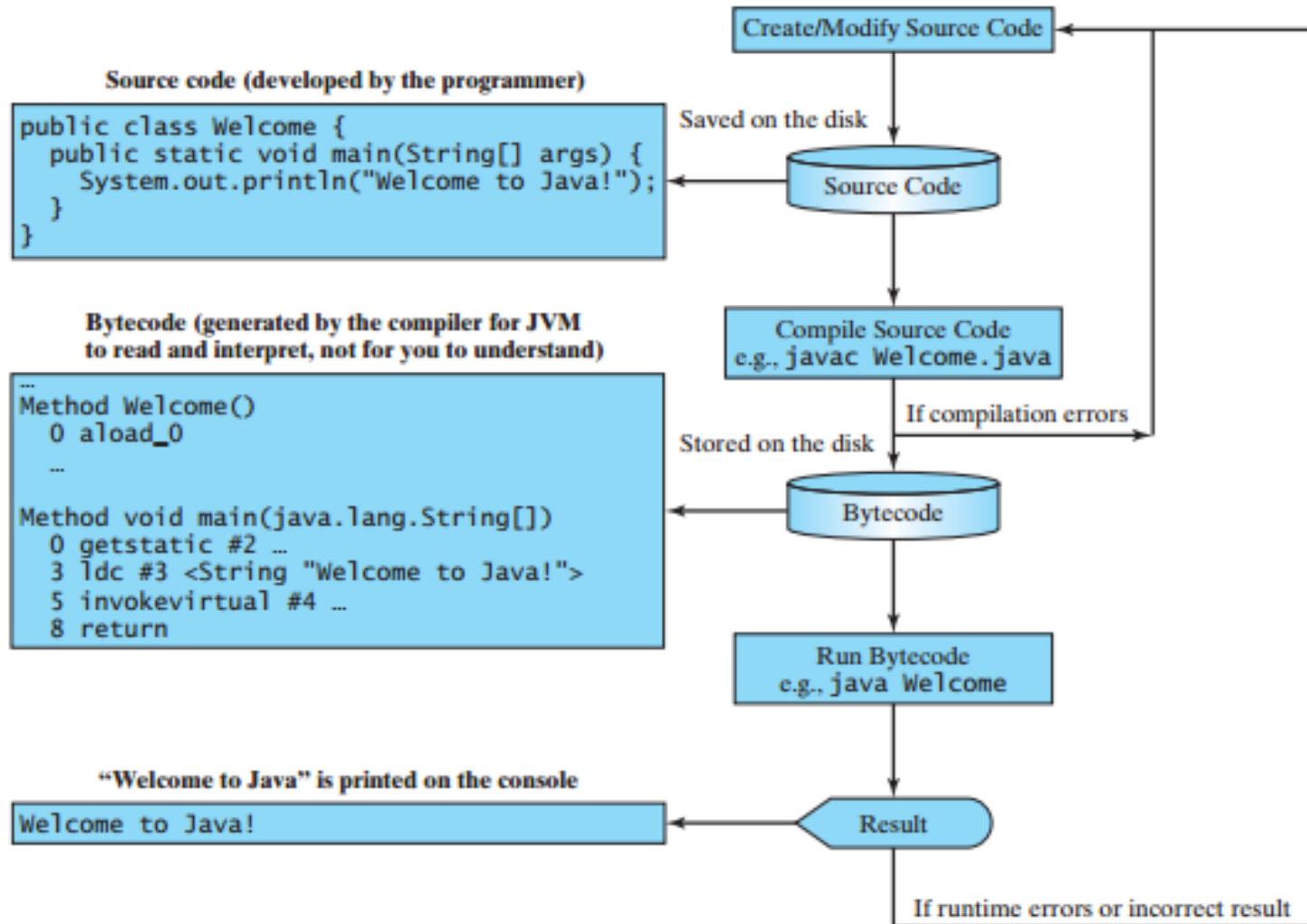


byte code

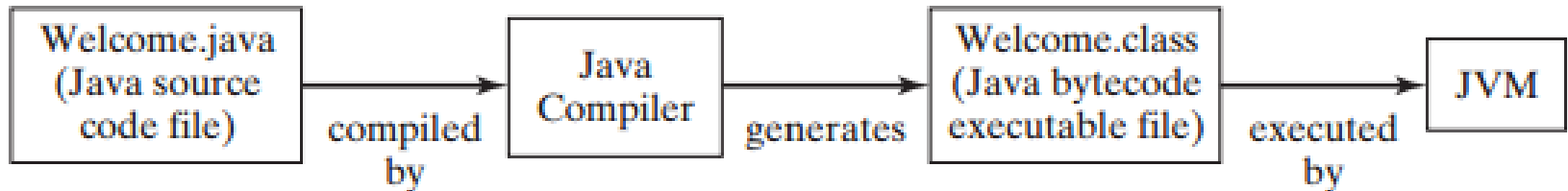
- hello.java
- hello.java => javac => hello.class [is not directly executable on OS]
- hello.class => Java Virtual Machine (software machine on top of machine)

- hello.c => after compilation => hello.exe --→ execute (Windows)
- hello.c -→ after compilation → hello.out → execute (Linux)
  
- hello.java → compile → hello.class → execute anywhere where there is JVM

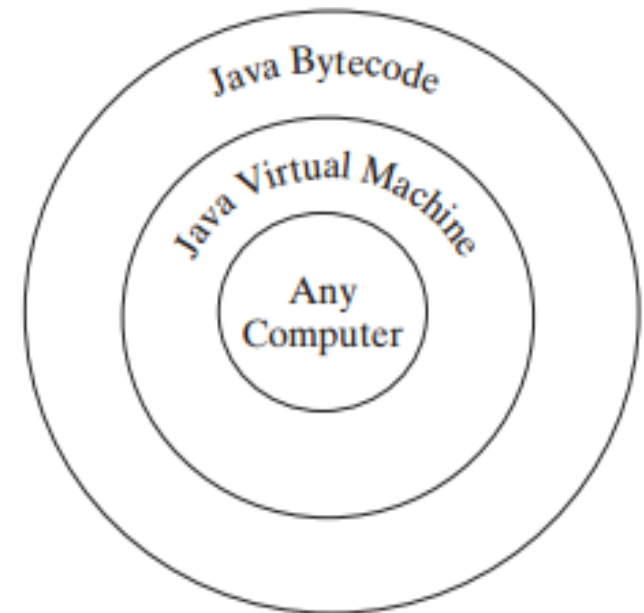
# Java Program Execution



# Java Bytecode and JVM



- ❑ Java source file using any text editor such as Eclipse, NetBeans, TextPad or Notepad++
- ❑ Compile the source codes `javac source.java`, and execute `java source`
- ❑ Integrated Development Environment (IDE): Netbeans / Eclipse
- ❑ JVM for every platform
- ❑ Compile once, run anywhere



# A Sample Java Program

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

- The name of the source file must be `Welcome.java`
- This program prints: `Welcome to Java!`

# Anatomy of a Java Program

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



# Smallest Java Program

```
public class Smallest
{
    public static void main(String[] args)
    {
    }
}
```

# Integrated Development Environment (IDE)

- Eclipse
- Command Prompt
  - `javac program_name.java`
  - `Java program_name`