JAVA™

AN INTRODUCTION TO

PROBLEM SOLVING

AND PROGRAMMING

WALTER SAVITCH

COMPUTER BASICS

Ch 1.1

Computer Basics: Outline

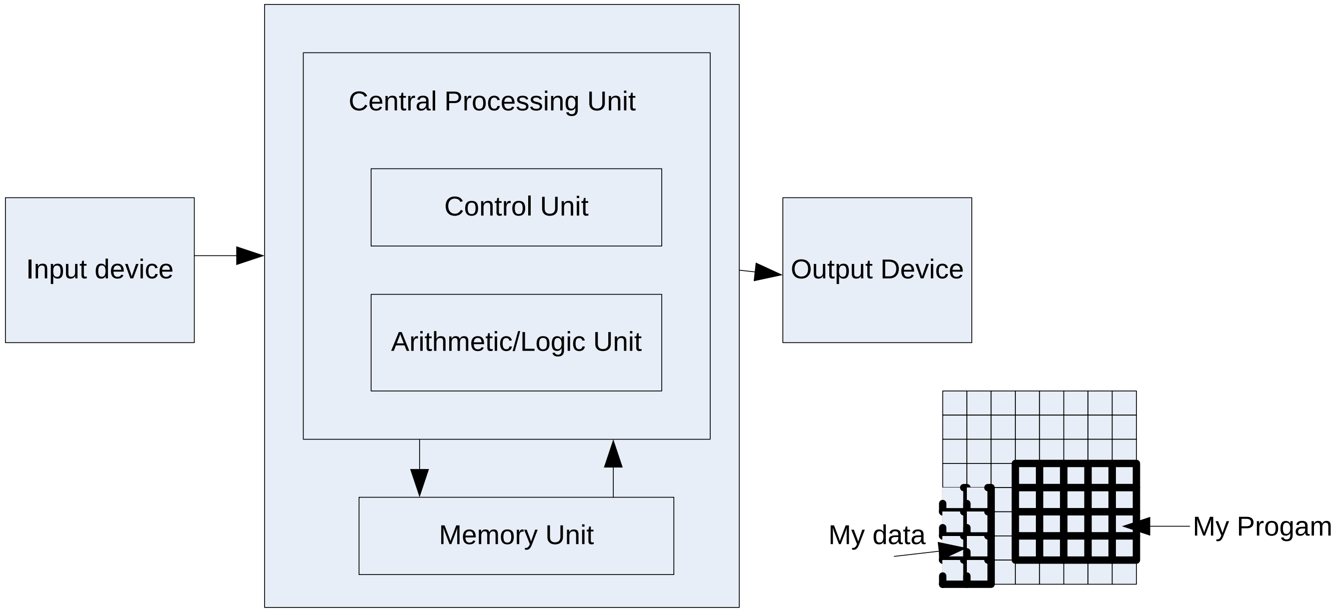
- Computer Components
- Programs
- Programming Languages and Compilers
- Java Byte-Code
- Graphics Supplement (FYI)

Computer Components

- Computer systems consist:
  - Hardware: the tangible parts of computer systems.
  - Software: includes programs - sets of instructions for the computer to follow.
- Generally, computers have similar components
  - Input devices (keyboard, mouse, etc.)
  - Output devices (display screen, printer, etc.)
  - A processor
  - Two kinds of memory (main memory and auxiliary memory).

The Processor

- It’s called the CPU (central processing unit):
  - Control unit (supervises the operation of other devices)
  - Arithmetic and Logic unit (ALU) (performs calculations)
- It processes a program’s instructions.
- It can process only very simple instructions.



The Memory

Main Memory

- Working memory used to store
  - The current program
  - The data the program is using
  - The results of intermediate calculations
- Usually measured in megabytes (e.g. 8 gigabytes of RAM)
  - RAM is short for random access memory
  - A byte is a quantity of memory

Auxiliary Memory

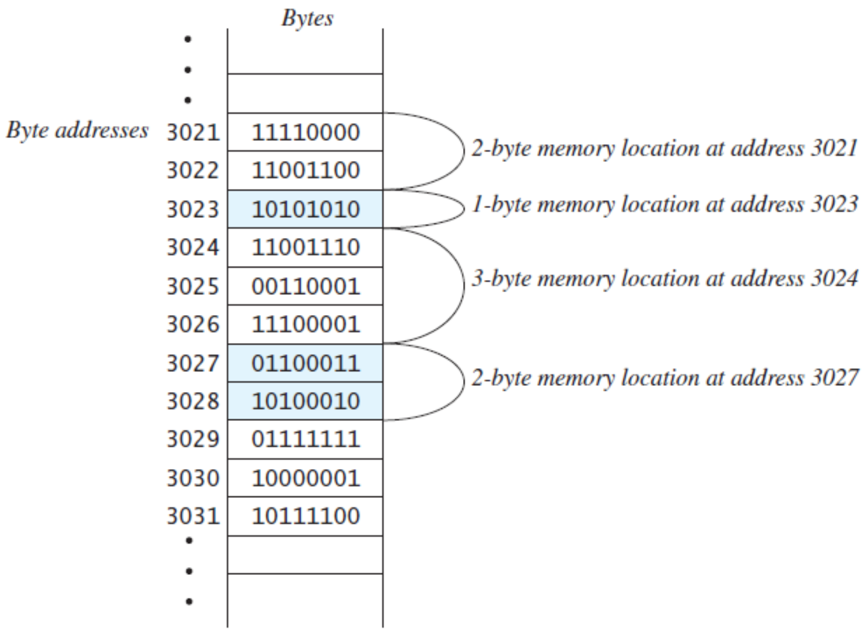
- Also called secondary memory
- Disk drives, CDs, DVDs, flash drives, etc.
- More or less permanent (nonvolatile)
- Usually measured in gigabytes (e.g. 50 gigabyte hard drive)

Bits & Bytes and Storing Data

- A bit is a digit with a value of either 0 or 1.
- A byte consists of 8 bits.
- Each byte in main memory resides at a numbered location called its address.

- Data of all kinds (numbers, letters, strings of characters, audio, video, even programs) are encoded and stored using 1s and 0s.
- When more than a single byte is needed, several adjacent bytes are used. The address of the first byte is the address of the unit of bytes.

FIGURE 1.1 Main Memory



Byte addresses	*Bytes*
3021	11110000
3022	11001100
3023	10101010
3024	11001110
3025	00110001
3026	11100001
3027	01100011
3028	10100010
3029	01111111
3030	10000001
3031	10111100

2-byte memory location at address 3021
1-byte memory location at address 3023
3-byte memory location at address 3024
2-byte memory location at address 3027

Files

- Large groups of bytes in auxiliary memory are called files.
- Files have names.
- Files are organized into groups called directories or folders.
- Java programs are stored in files.
- **Programs files are copied from auxiliary memory to main memory in order to be run**.

Programs

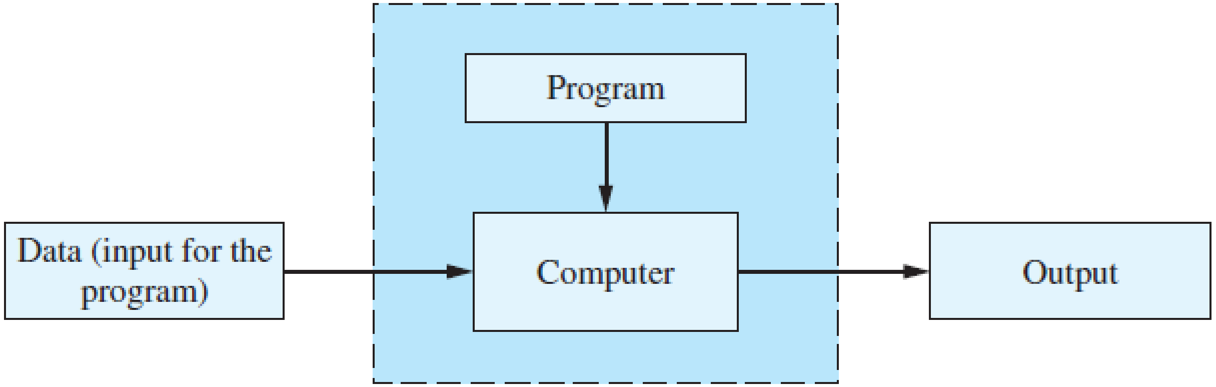
Any thoughts on the use of the word “almost” here?

- A program is a set of instructions for a computer to follow.
- We use programs almost daily (email, word processors, video games, bank ATMs, etc.).
- Following the instructions is called running or executing the program.

Running a program: Input and Output

- This is how we think of running a program:

FIGURE 1.2 Running a Program



- Strictly speaking, **the program is also an input** to the computer in addition to the data needed by the program
- The output is the result(s) produced by following the instructions in the program.

The Operating System (OS)

- The operating system is a supervisory **program** that oversees the operation of the computer.
- The operating system retrieves and starts other programs for you.
- Well-known operating systems including: Microsoft Windows, Apple’s Mac OS, Linux, and UNIX.

Programming Languages

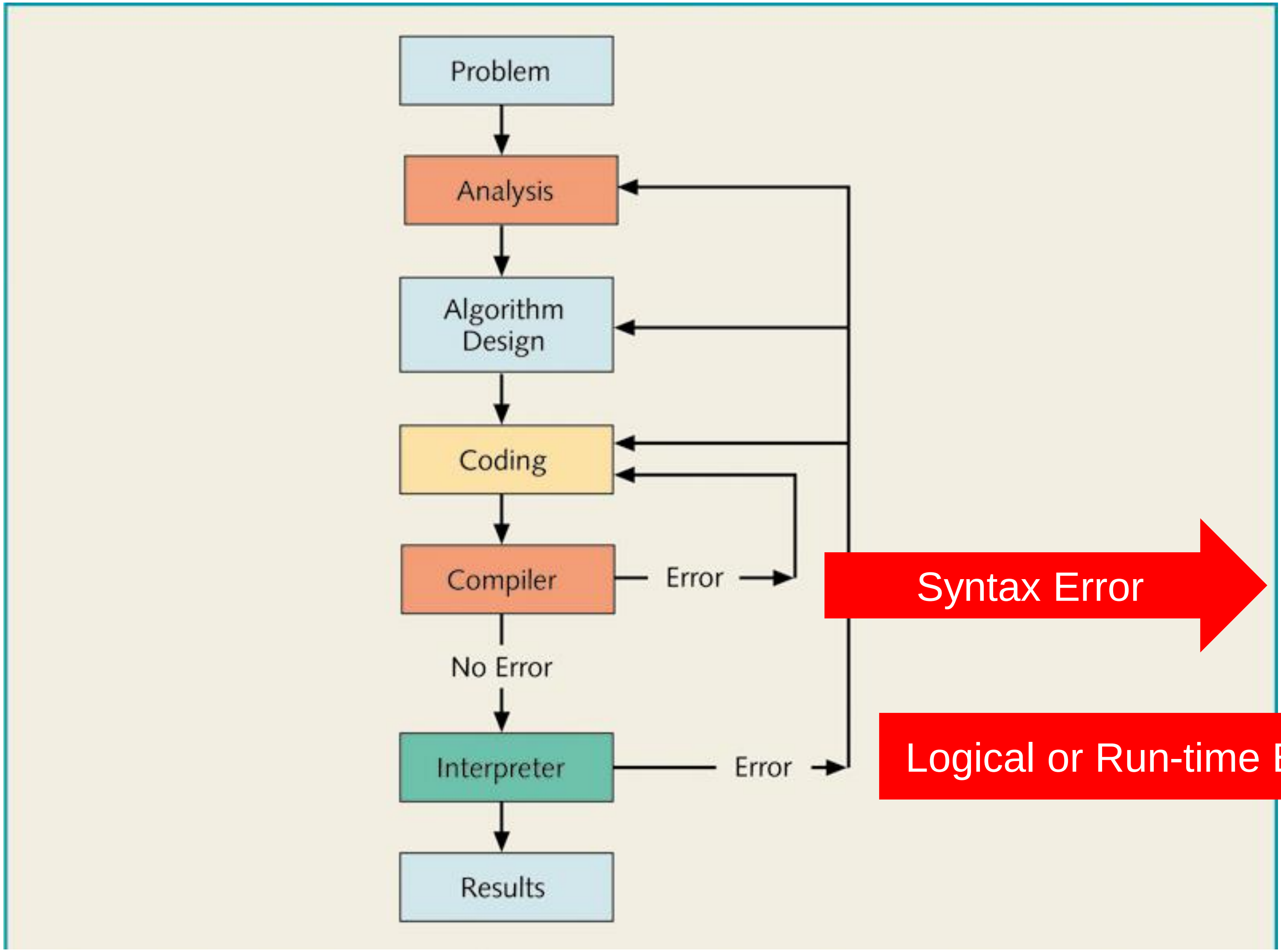
- High-level languages are relatively easy to use
  - Java, C#, C++, Visual Basic, Python, Ruby.
- computer hardware does not understand high-level languages.
- A compiler translates a program from a high-level language to a low-level language the computer can run.

Program Solving Cycle

- Solving a program goes through the following main steps:

1. Analysis: Outline the solution requirements
2. Design an appropriate algorithm or a flowchart.
3. Code the solution in a high programming language (such as Java)
4. Compile the code into machine language. Verify that the program works:
    1. If there is an error, correct it by going to step 3.
    2. If there is no error, proceed to step 5.
5. Run the program. Verify the results:
    1. If the output does not give the required results, go to step 1.
    2. If the output matches the required result, you are done.

Program Solving Cycle (cont’d)



Syntax Error

Logical or Run-time Error

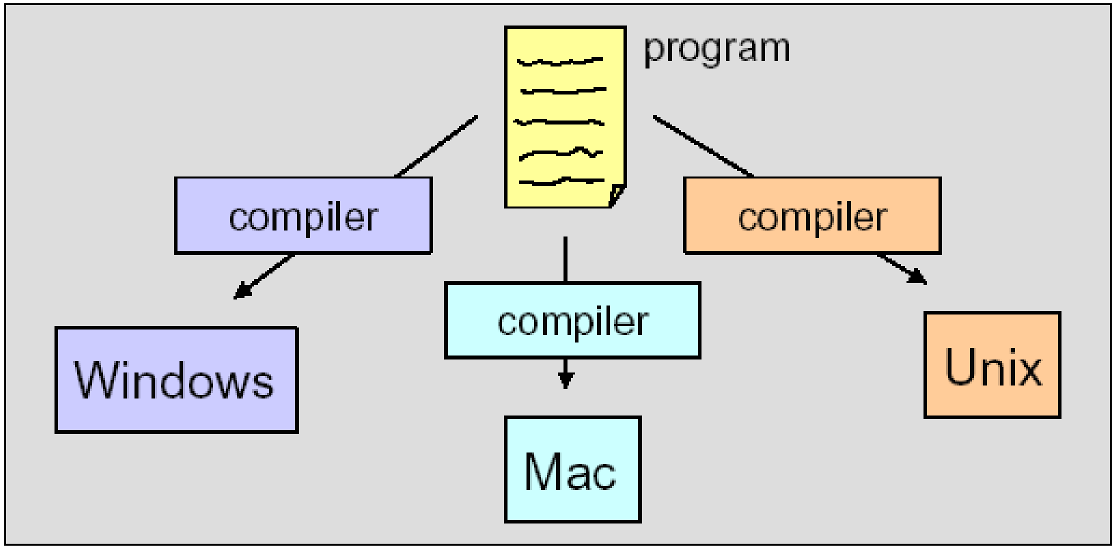
Program Compilation

- The compiler:
  - is a software that checks the correctness of the source code according to the language rules
  - If it produces an error, this is called a syntax error.
  - it translates the source code into a machine code if no errors were found.

Platform dependency

- Machine code depends on the computer hardware: we say that the compiled version is platform-dependent. For example:
  - a program compiled on a machine that works under the Windows OS, **cannot** run on another machine that works under the MAC OS.
  - In this case, the program should be re-compiled under the MAC operating system.
- However, **Java is platform-independent**:
  - In other words, a Java program that is compiled under the Windows OS **can** run under the MAC OS without being re-compiled.

Program Compilation (cont'd)

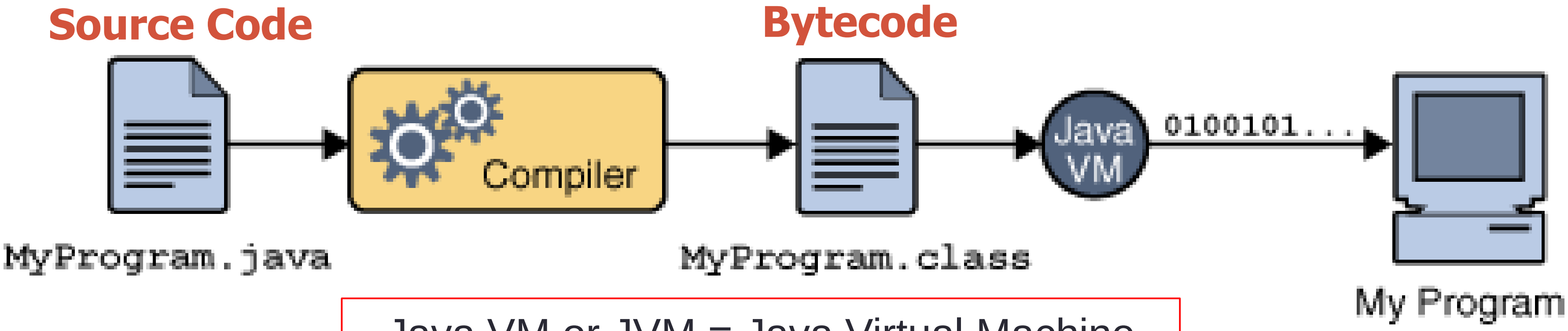


Platform-Dependent Compilation

- Note:
  - Most compilers are very large programs that are expensive to produce

Java Byte-Code

- The Java compiler does not translate a Java program into machine language for a particular computer.
- Instead, it translates a Java program into byte-code.
  - Byte-code is the machine language for a hypothetical computer (or interpreter) called the Java Virtual Machine.



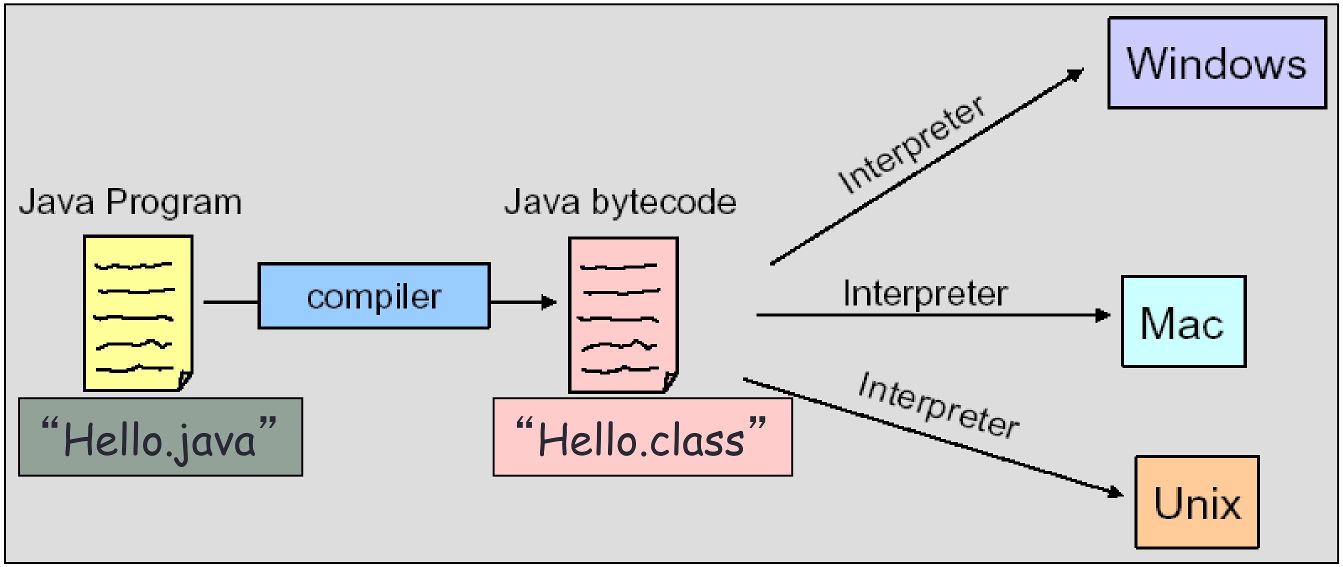
Java VM or JVM = Java Virtual Machine

Java Byte-Code

- A byte-code program is easy to translate into machine language for any particular computer.
- A program called an interpreter translates each byte-code instruction, executing the resulting machine-language instructions on the particular computer before translating the next byte-code instruction.
- Most Java programs today are executed using a Just-In-Time or JIT compiler in which byte-code is compiled as needed and stored for later reuse without needing to be re-compiled.

Portability

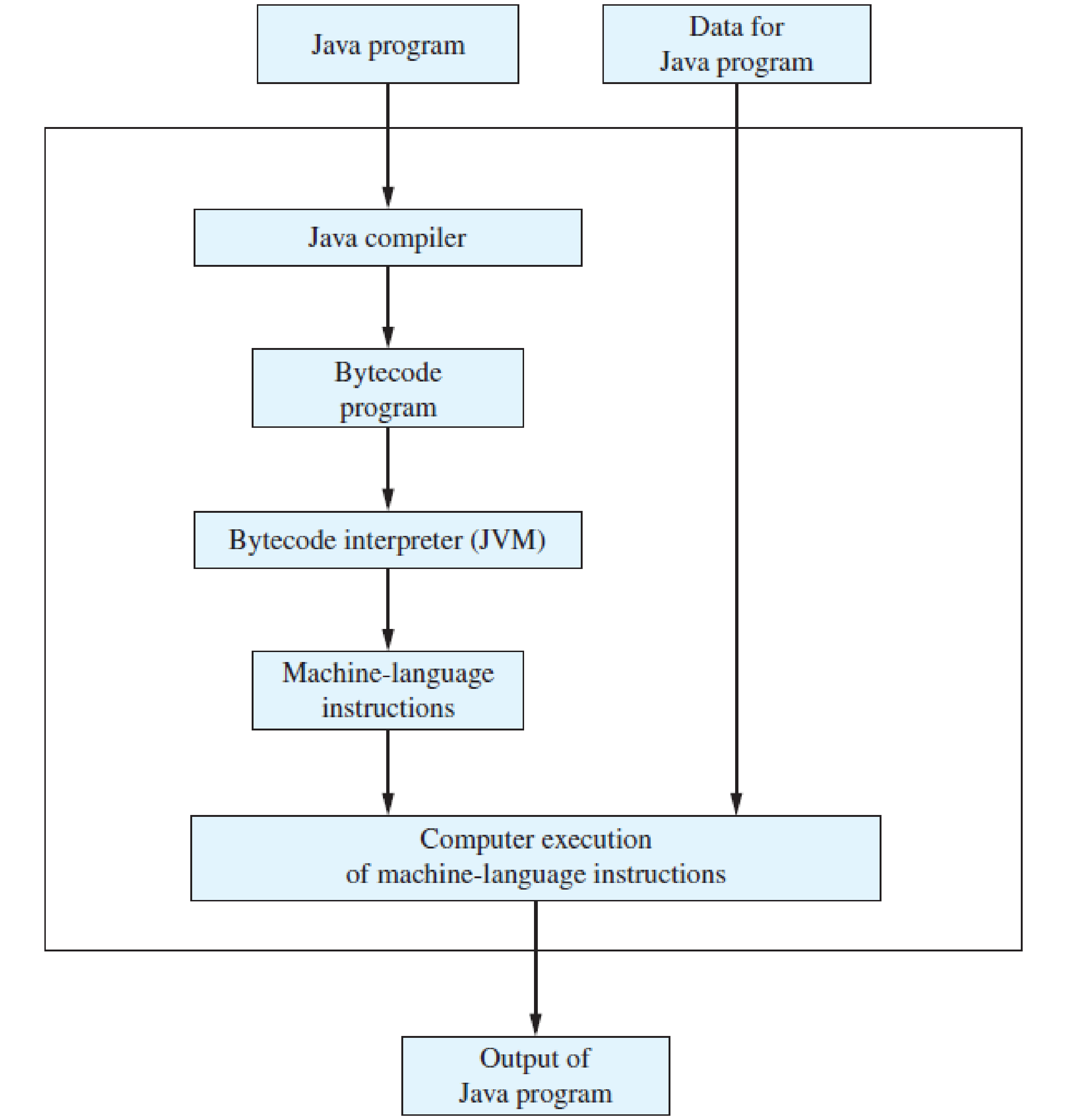
- The byte-code can be used on any computer with a byte-code interpreter and without a need to recompile.
- Byte-code can be sent over the Internet and used anywhere in the world.
- This makes Java suitable for Internet applications.



Class Loader

- A Java program typically consists of several pieces called classes.
- Each class may have a separate author and each is compiled (translated into byte-code) separately.
- A class loader (called a linker in other programming languages) automatically connects the classes together.

FIGURE 1.3 Compiling and Running a Java Program


Java program
Data for Java program
Java compiler
Bytecode program
Bytecode interpreter (JVM)
Machine-language instructions
Computer execution of machine-language instructions
Output of Java program