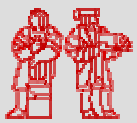
Lecture 1: Introduction to Java

MIT-AITI Kenya 2005

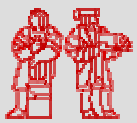
Lecture Outline

- What a computer program is
- How to write a computer program
- The disadvantages and advantages of using Java
- How a program that you write in Java is changed into a form that your computer can understand
- Sample Java code and comments

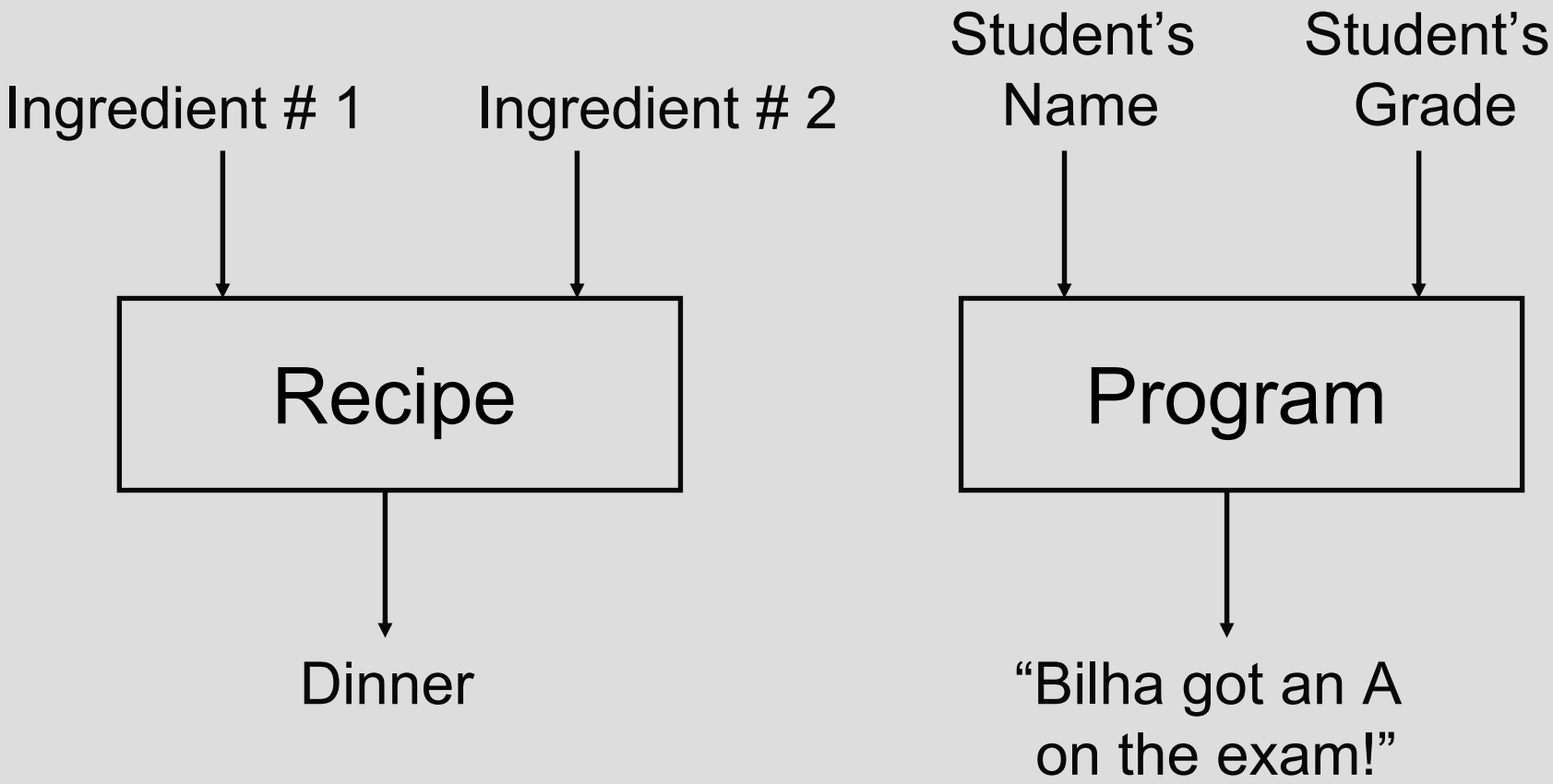


Computer Program vs. Food Recipe

Food Recipe		Computer Program
A *chef* writes a set of instructions called a *recipe*	⟷	A *programmer* writes a set of instructions called a *program*
The recipe requires specific *ingredients*	⟷	The program requires specific *inputs*
The *cook* follows the instructions step-by-step	⟷	The *computer* follows the instructions step-by-step
The *food* will vary depending on the *amount of ingredients* and the *cook*	⟷	The *output* will vary depending on the *values of the inputs* and the *computer*



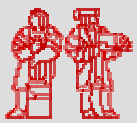
Recipe and Program Examples



What is a computer program?

- For a computer to be able to perform specific tasks (i.e. print what grade a student got on an exam), it must be given instructions to do the task

- The set of instructions that tells the computer to perform specific tasks is known as a *computer program*

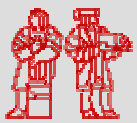


Writing Computer Programs

- We write computer programs (i.e. a set of instructions) in programming languages such as C, Pascal, and Java

- We use these programming languages because the are easily understood by humans

- But then how does the computer understand the instructions that we write?

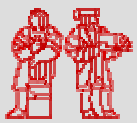


Compiling Computer Programs

- Computers do not understand programs written in programming languages such as Pascal and Java

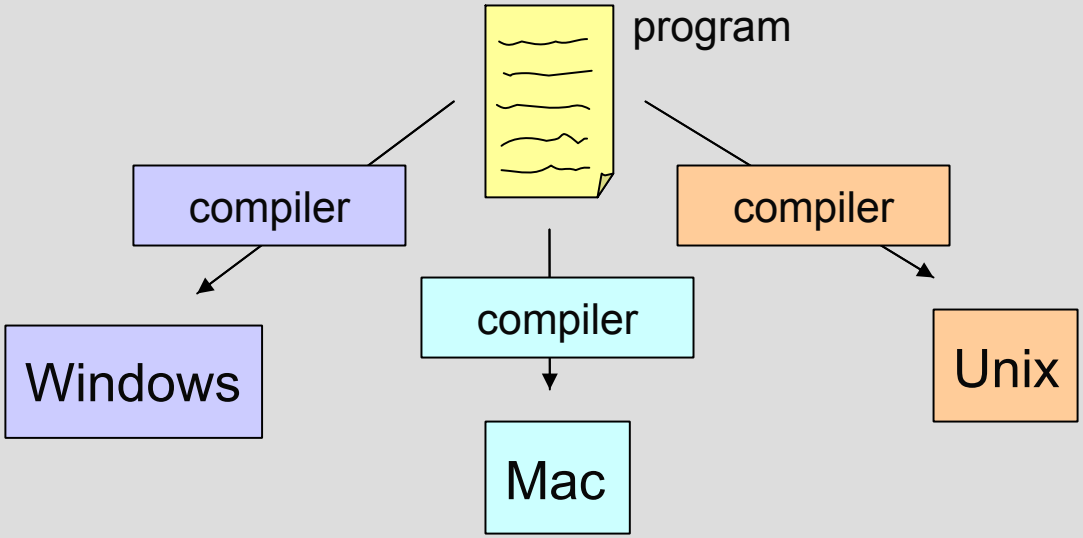
- Programs must first be *compiled* or converted into machine code that the computer can run

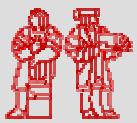
- A program that translates a programming language into machine code is called a *compiler*



Compiling Computer Programs

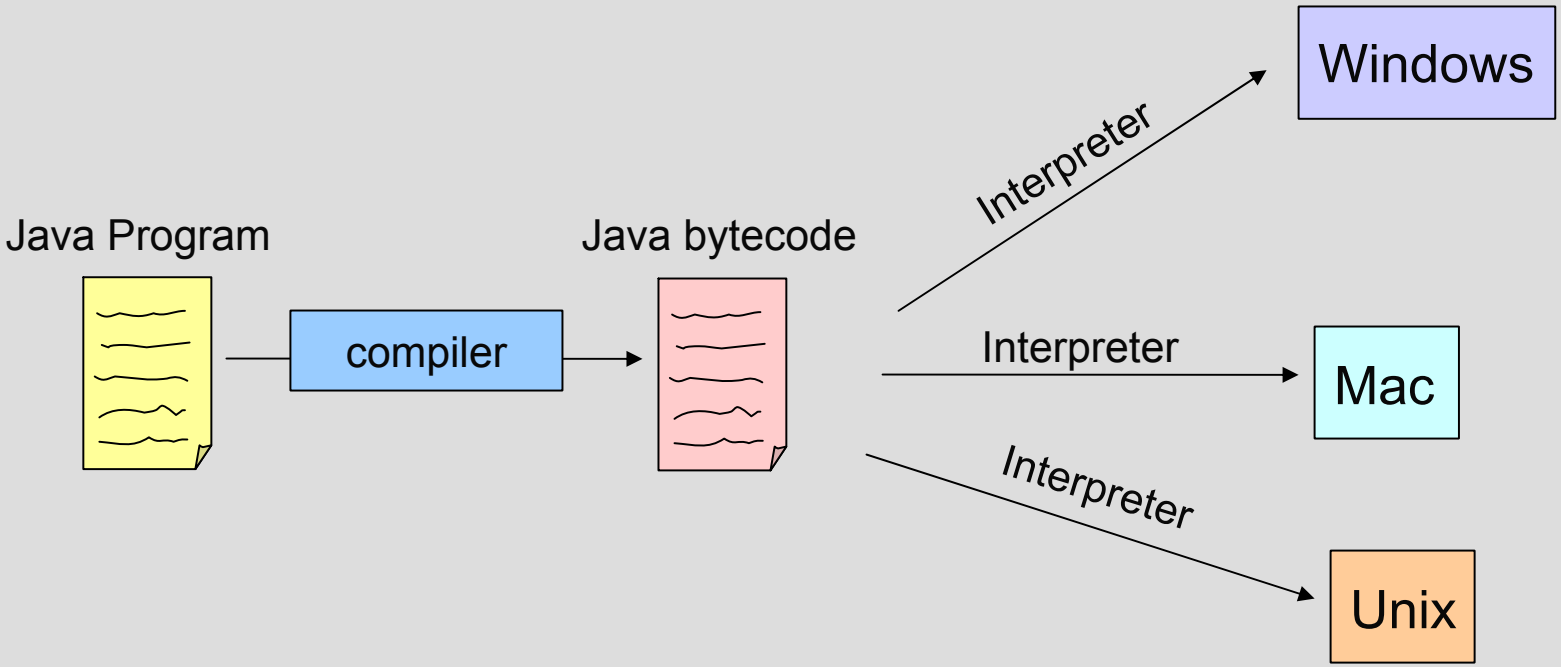
- Because different *platforms*, or hardware architectures along with the operating systems (Windows, Macs, Unix), require different machine code, you must compile most programs separately for each platform.

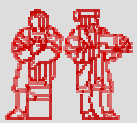




Compiling Java Programs

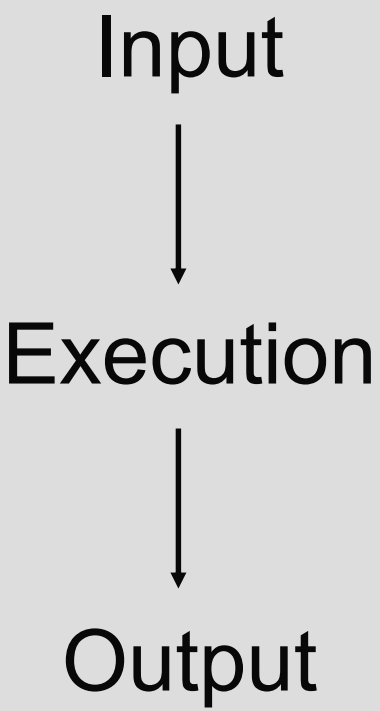
- The Java compiler produces *bytecode* not machine code
- Bytecode is converted into machine code using a *Java Interpreter*
- You can run bytecode on any computer that has a Java Interpreter installed





Running Programs

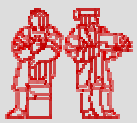
Most programs follow a simple format:



An *input* is something you put into your program that your program can use during its execution.

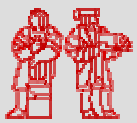
When you *execute* your program, the computer evaluates the instructions in your program step-by-step.

An *output* is something your program produces after execution.

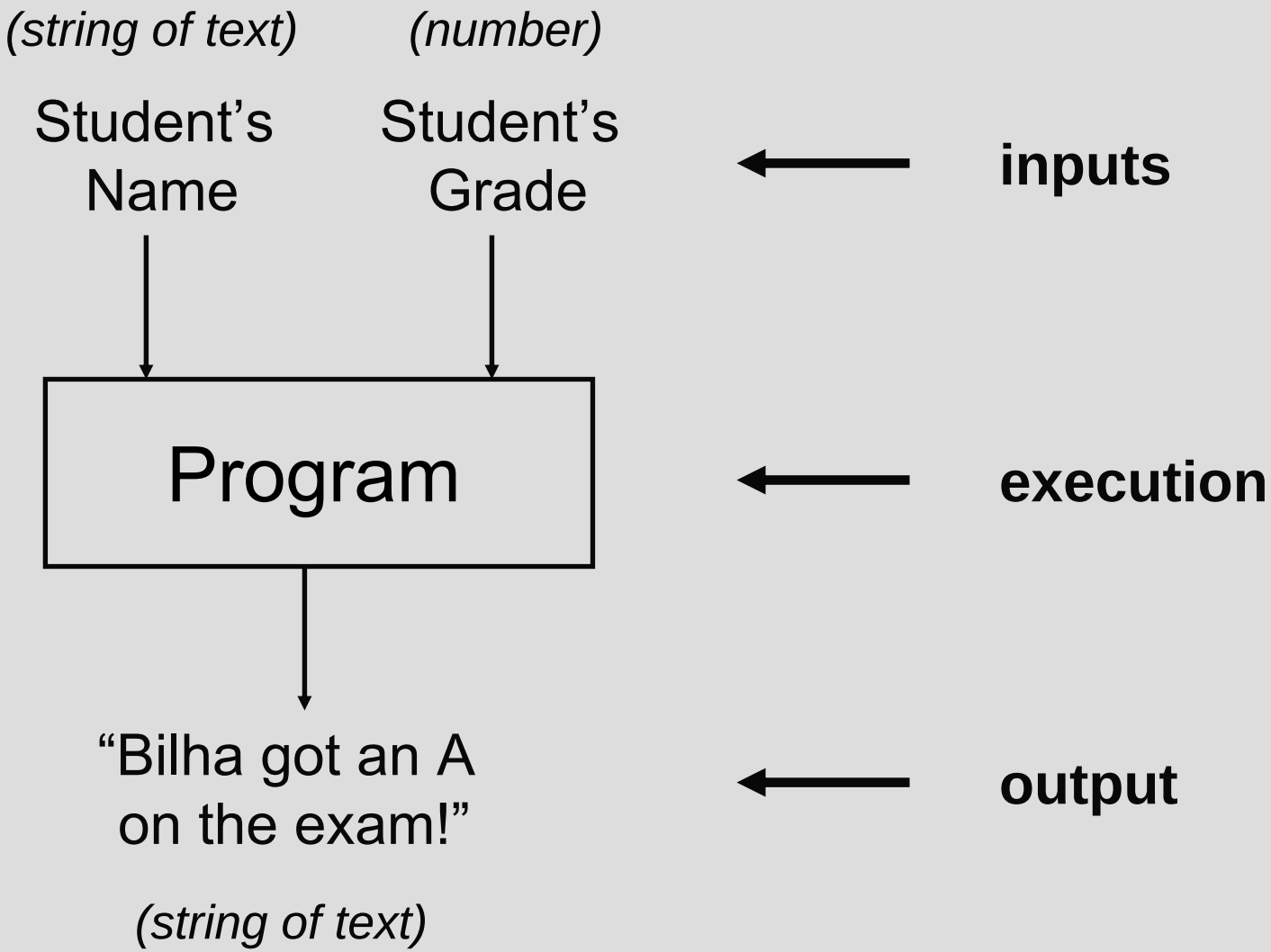


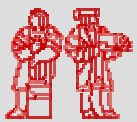
Running Programs

- Inputs
  - Can come from many sources, such as users, files, and other programs
  - Can take on many forms, such as text, graphics, and sound
- Outputs
  - Can also take on many forms, such as numbers, text, graphics, sounds, or commands to other programs



Running Programs

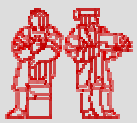




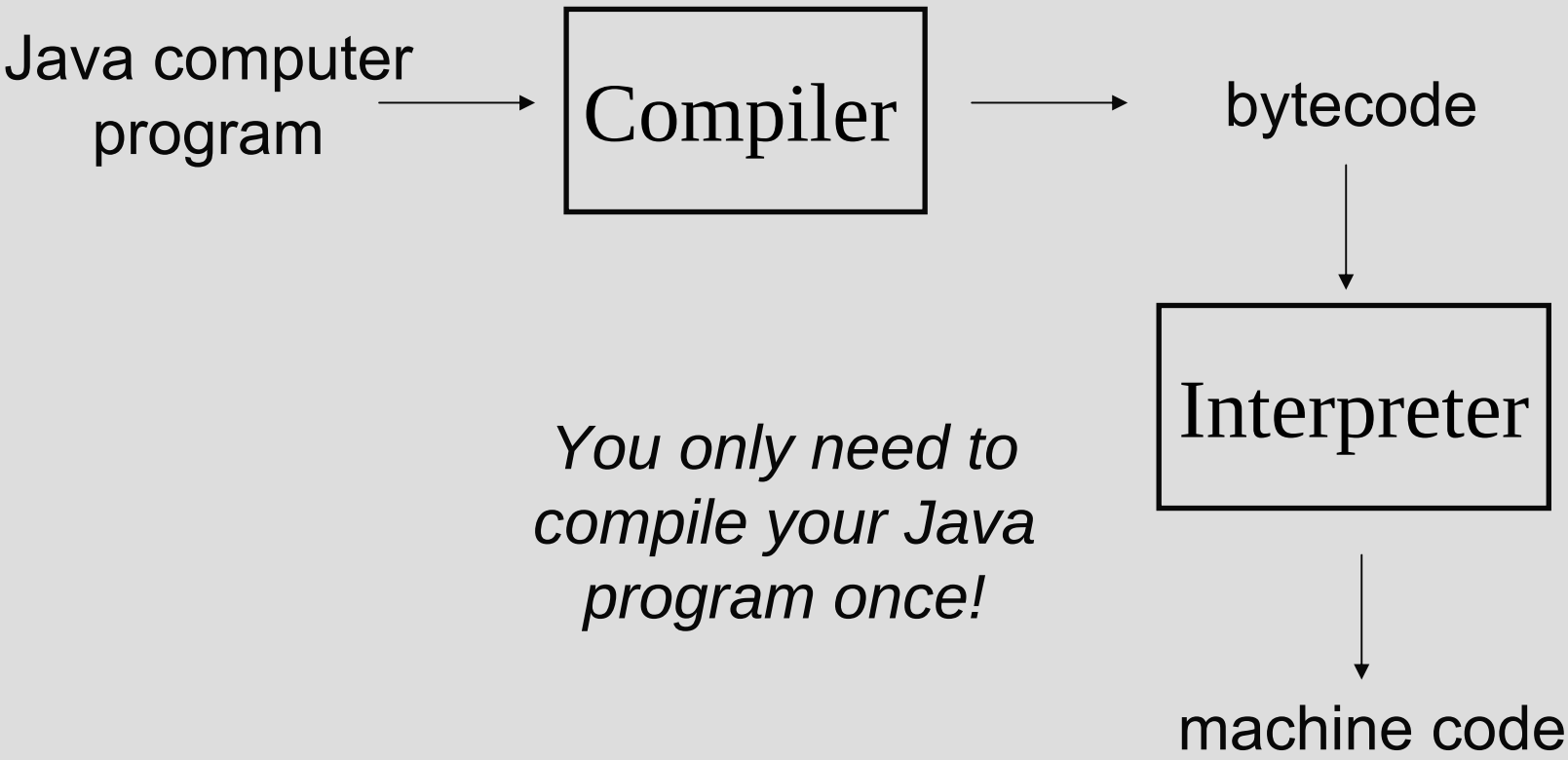
Running Programs Pop Quiz

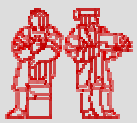
You want to write a program
that adds two numbers

- What are the inputs to your program?
  the two numbers
- What does your program do?
  adds the two numbers
- What is the output of your program?
  the sum of the two numbers



Compiling and Running Programs in Java



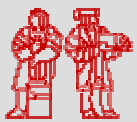


Advantages of Using Java

- Once a Java program is compiled you can run the bytecode on any machine with a Java interpreter. Because you do not have to recompile the program for each machine, Java is *platform independent.*

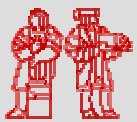
- Java is safe. In other words, the Java language and compiler prevent certain common programming bugs and dangerous operations

- Java standardizes many useful operations such as managing network connections and providing graphical user interfaces



Disadvantages of Using Java

- Running bytecode through an interpreter is not as fast as running machine code, which is specific to each platform.
- Using platform specific features (e.g. Windows taskbar) is difficult in Java because Java is platform-independent.
- In order to run a Java program on multiple machines, you must install a Java Interpreter on each machine



Running and Compiling Programs Pop Quiz

- How many times do you need to compile a Java program in order to run it on two different platforms? Why?

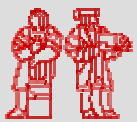
  Once! Java is platform independent.

- When you compile a Java program, what is the name of the resulting output?

  Bytecode

- Why might running a program in Java be slower than running a program in a language that requires a separate compiler?

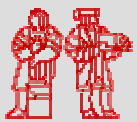
  Bytecode has to be run through the interpreter. This process takes more time that running machine code that is specific to each platform.



Example Java Code

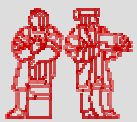
- This is part of the code that you will be writing in Lab 0:

```
/* The HelloWorld class prints “Hello,
World!” to the screen */
public class HelloWorld {
    public static void main(String[] args) {
        // Prints “Hello, World!”
        System.out.println("Hello, World!");
    }
}
```



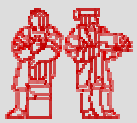
Comments

- *Comments* are used to describe what your code does as an aid for you or others reading your code. The Java compiler ignores them.
- Comments are made using `//`, which comments to the end of the line, or `/* */`, which comments everything inside of it (including multiple lines)
- Two example comments:
  - `/* The HelloWorld class prints “Hello, World!” to the screen */`
  - `// Prints “Hello, World!”`



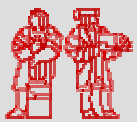
Comments on Commenting

- You may collaborate on software projects with people around the world who you'll never meet.
- Should be able to figure out how code works by reading comments alone.
- Anything that is not self-evident needs a comment.
- 50% of your code might be comments.
- Coding is easy; commenting is not.



This lecture covered…

- What a computer program is
- How to write a computer program
- How a program written in Java is changed into a form that a computer can understand
- The disadvantages and advantages of using Java
- An example of Java code and comments



EC.S01 Internet Technology in Local and Global Communities
Spring 2005-Summer 2005