CIS 22A – Lecture 1

Manish Goel

Why Program?

Computer – programmable machine designed to follow instructions

Program – instructions in computer memory to make it do something

Programmer – person who writes instructions (programs) to make computer perform a task

Things to remember

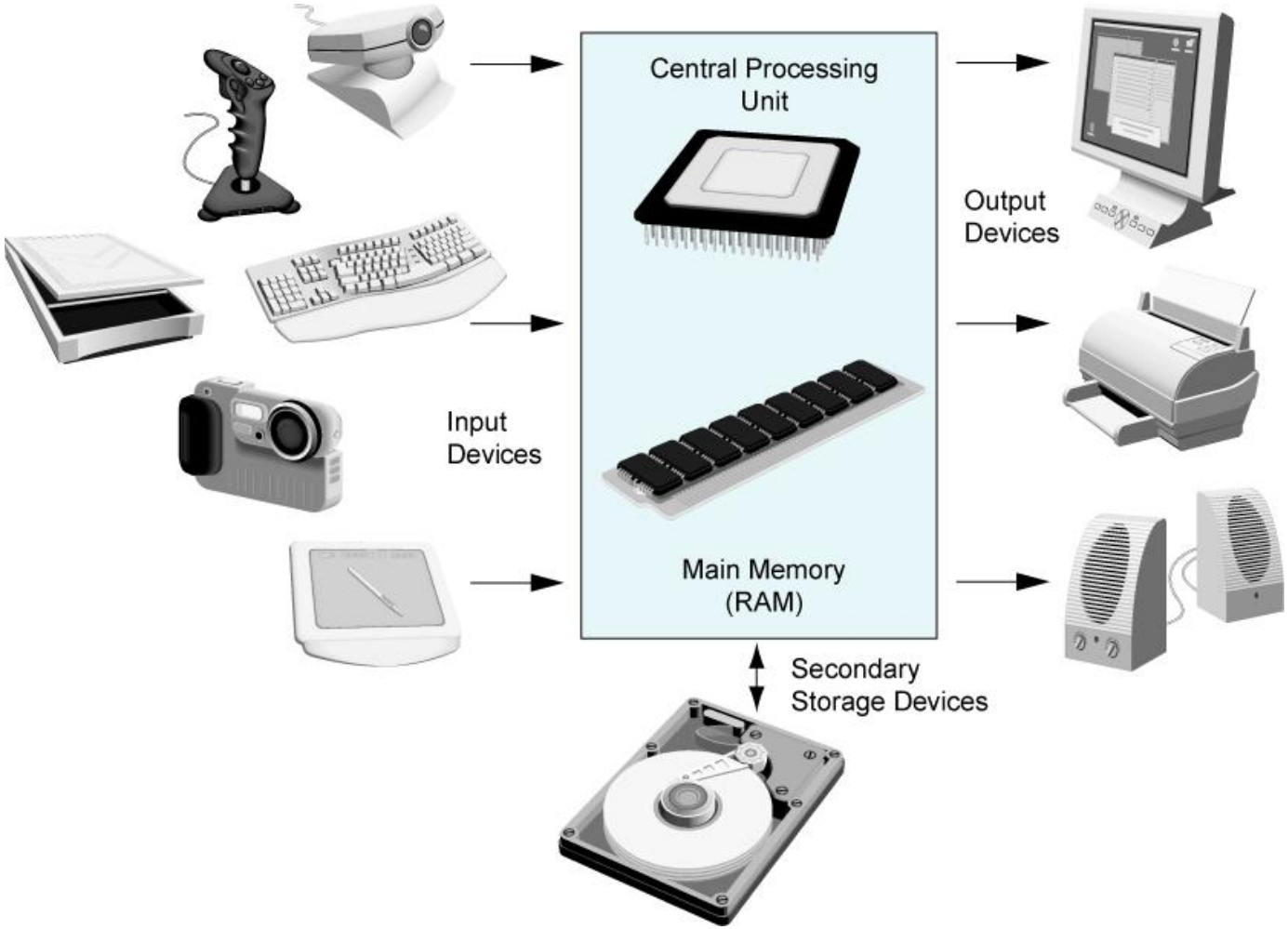
Without programmers → no programs

Without programs → a computer is useless

However:

- A computer has a very limited brain for logic
- It cannot make inferences or judgments
- It only does what it is told to do
- Thus, a programmer needs to think like a computer

What makes a Computer



How does a COMPUTER compute?

- The most necessary parts of the computer are the CPU and Main Memory.
- I/O devices like disk, mouse etc are needed for humans to interact with the computers.
- CPU is made up of a General Control Unit and the Arithmetic Logic Unit.
- The Main Memory is akin to a “Mailboxes Etc.” with addresses for each memory location (mailbox) – the addresses allow the CPU to place data or commands at specific memory locations.

Main Hardware Component Categories:

1. Central Processing Unit (CPU)
2. Main Memory
3. Secondary Memory / Storage
4. Input Devices
5. Output Devices

Central Processing Unit (CPU)

Comprised of:

- Control Unit
  - Retrieves and decodes program instructions
  - Coordinates activities of all other parts of computer
- Arithmetic & Logic Unit
  - Hardware optimized for high-speed numeric calculation
  - Hardware designed for true/false, yes/no decisions

CPU Organization

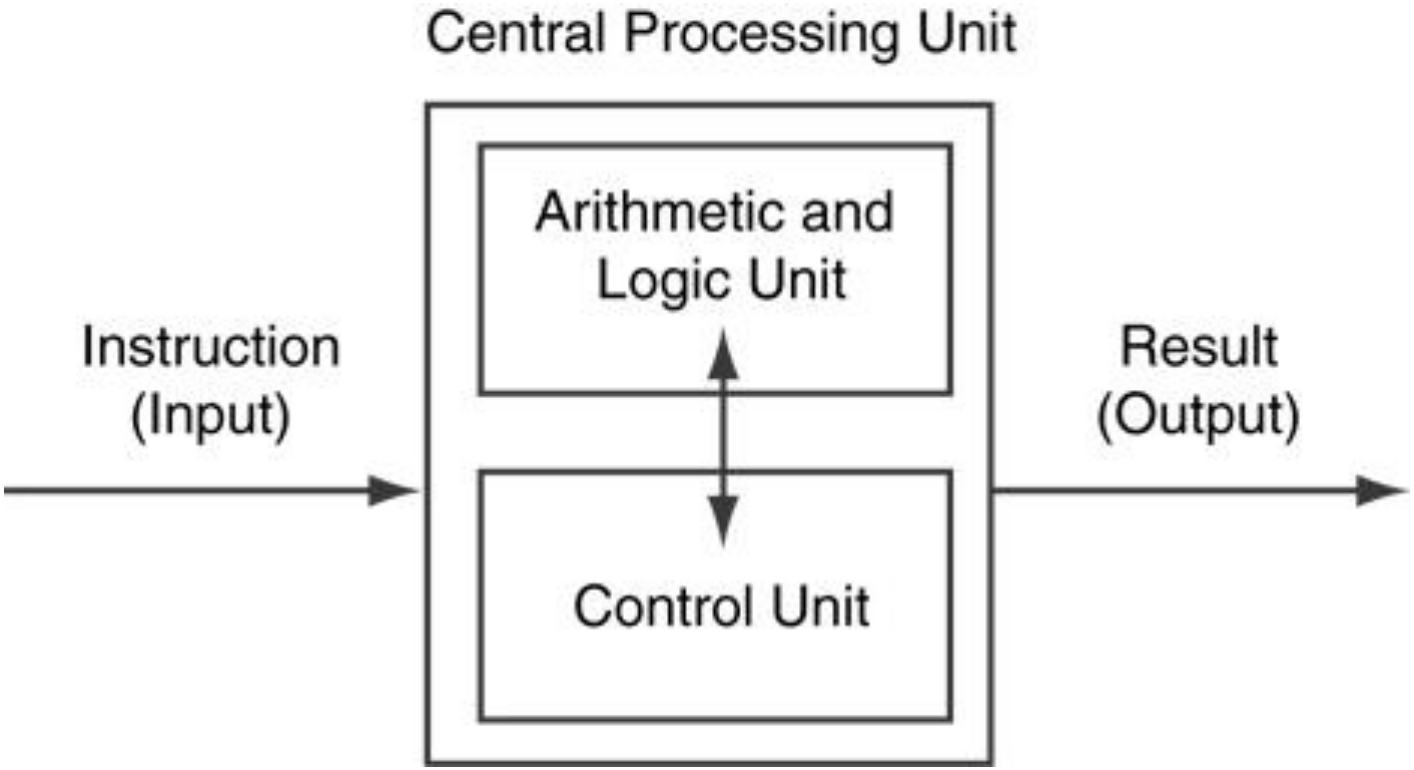


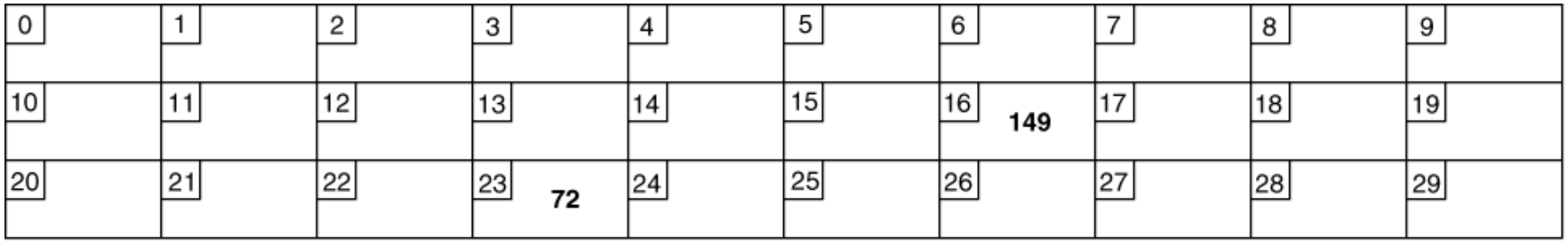
Figure 1-3

Main Memory

- It is volatile. Main memory is erased when program terminates or computer is turned off
- Also called Random Access Memory (RAM)
- Organized as follows:
  - bit: smallest piece of memory. Has values 0 (off, false) or 1 (on, true)
  - byte: 8 consecutive bits. Bytes have addresses.

Main Memory

- Addresses – Each byte in memory is identified by a unique number known as an *address*.



- In Figure 1-4, the number 149 is stored in the byte with the address 16, and the number 72 is stored at address 23.

Secondary Storage

- Non-volatile: data retained when program is not running or computer is turned off
- Comes in a variety of media:
  - magnetic: floppy disk, hard drive
  - optical: CD-ROM, DVD
  - Flash drives, connected to the USB port

Input Devices

- Devices that send information to the computer from outside
- Many devices can provide input:
  - Keyboard, mouse, scanner, digital camera, microphone
  - Disk drives, CD drives, and DVD drives

Computing Evolution

Mainframe

Mini

Micro (PC)

Laptop

Tablet

Smartphone

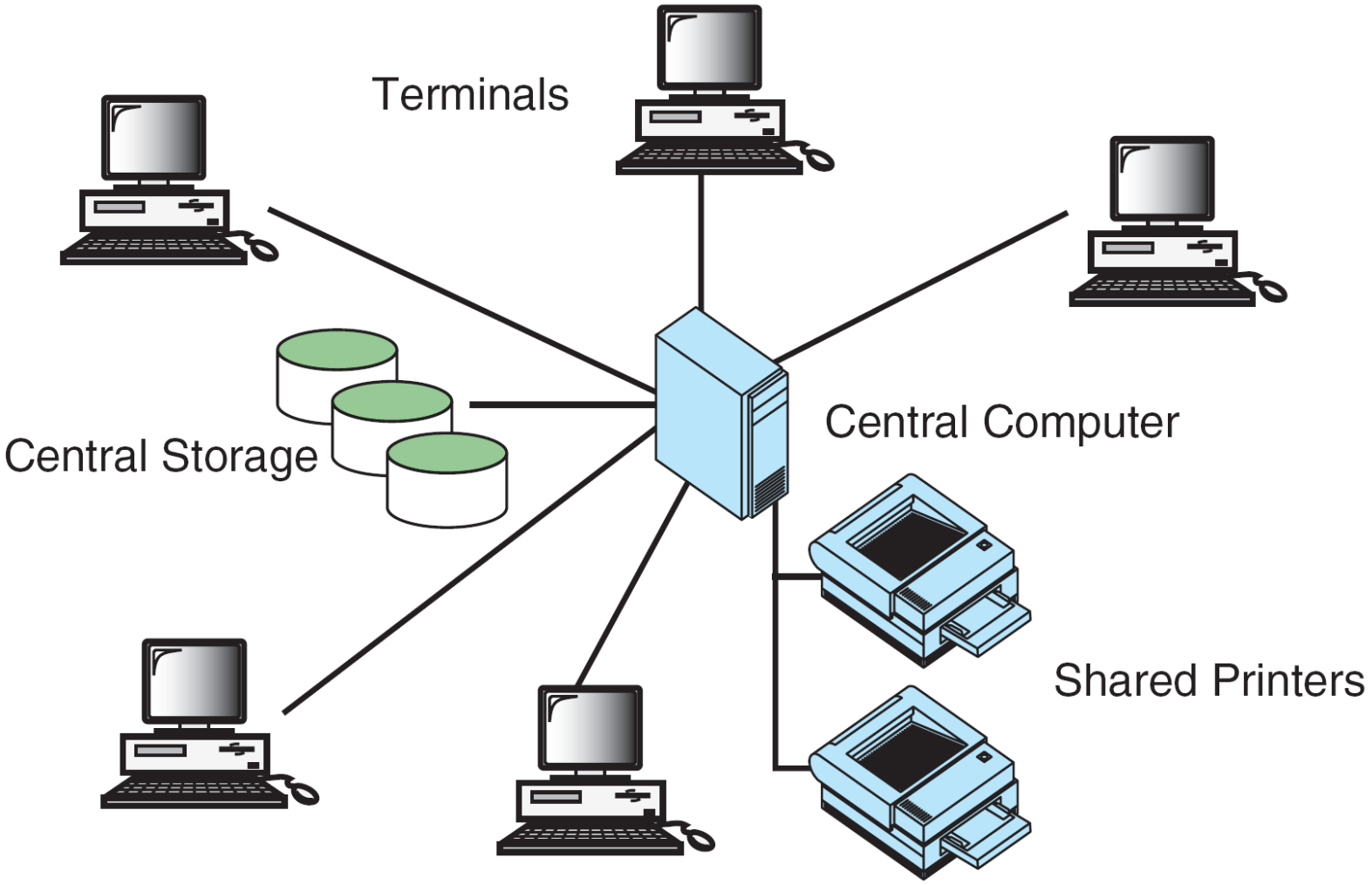
Moore’s Law

- Simply stated –

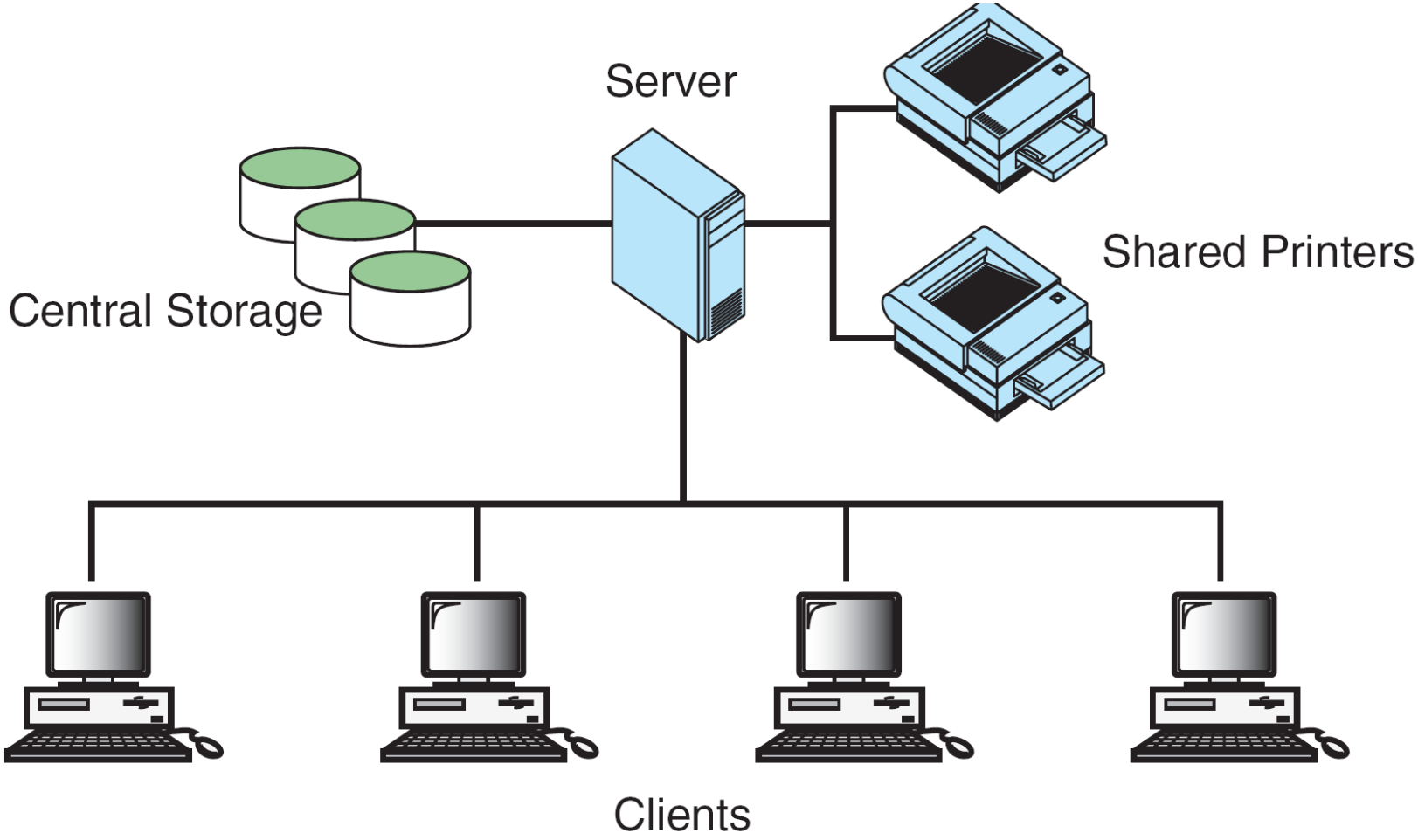
  “Computing power will double while computing cost will halve every two years”.

  Moore’s Law explains the journey from massive mainframes of the ‘50s to today’s smartphones.

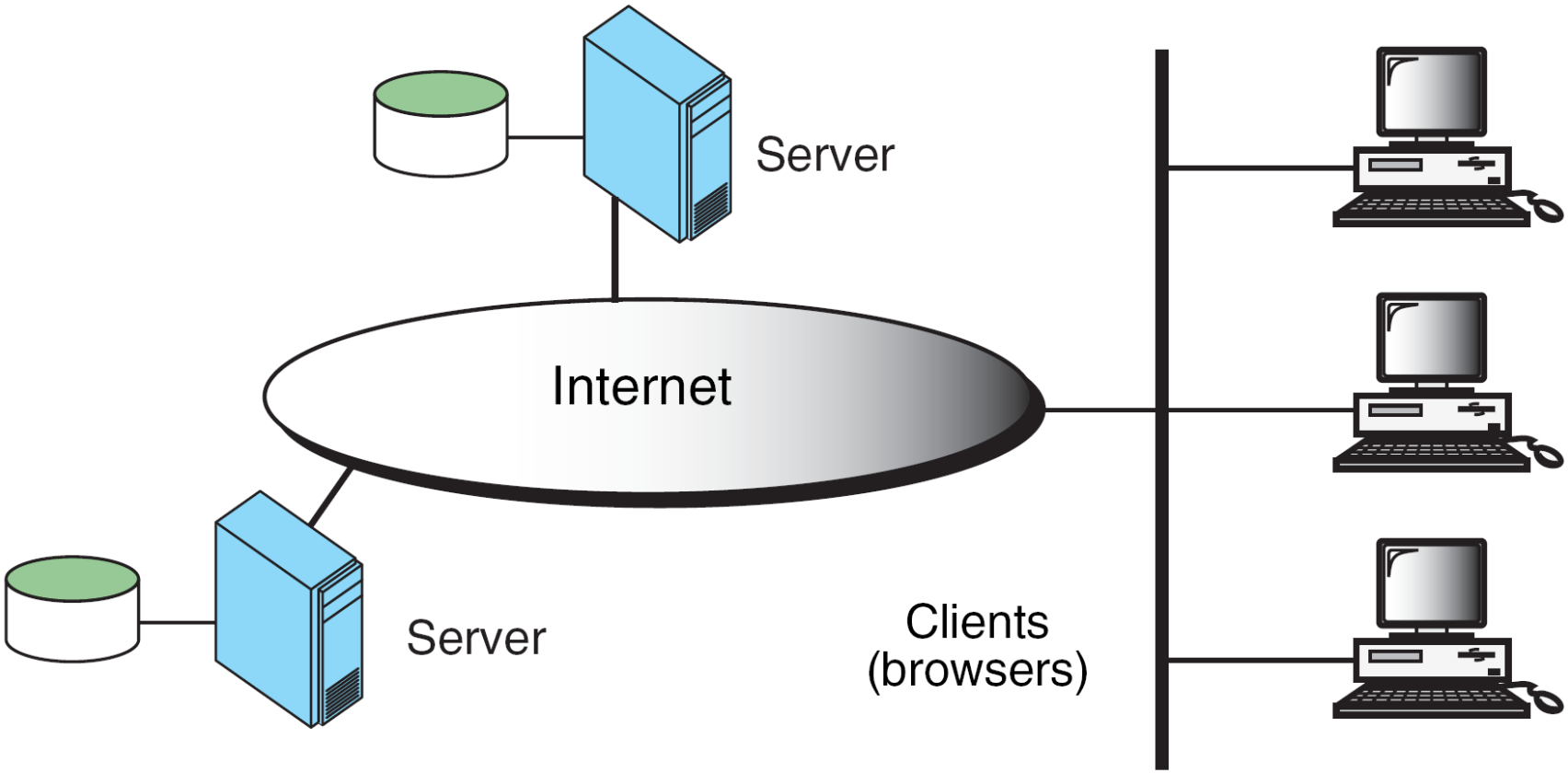
  Q: What really is “boot”-ing?



Time-sharing Environment



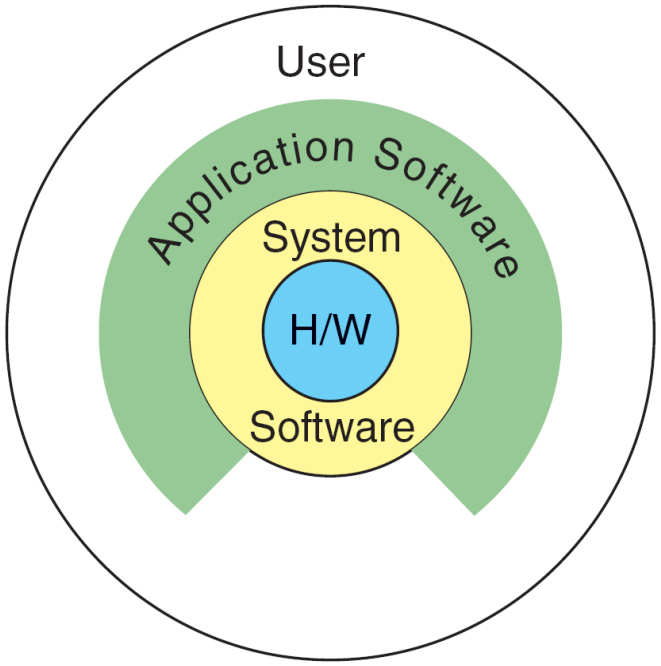
The Client/Server Environment



Distributed Computing

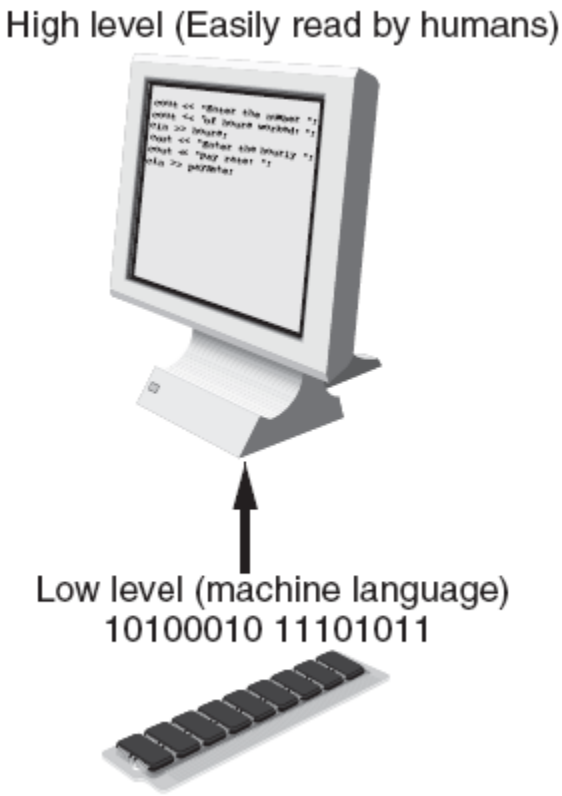
Software is a Program that runs on a Computer

- System software: programs that manage the computer hardware and the programs that run on them. *Examples*: operating systems, utility programs, software development tools
- Application software: programs that provide services to the user. *Examples* : word processing, games, programs to solve specific problems



Programs and Programming Languages

- Types of languages:
  - Low-level: used for communication with computer hardware directly. Often written in binary machine code (0's/1's) directly.
  - High-level: closer to human language
  - Assembly-level: Initial evolution of human readable languages



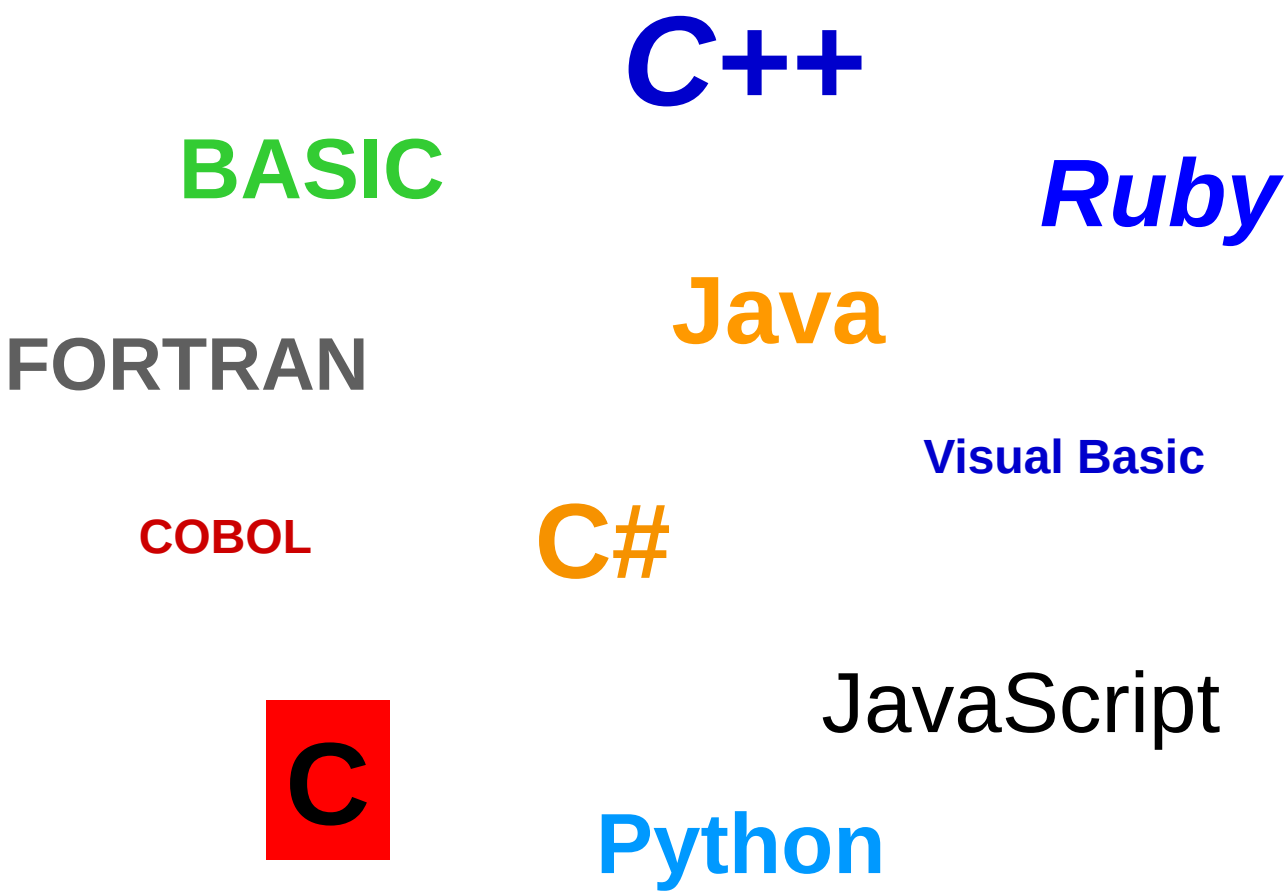
Machine Language

- Machine language instructions are binary numbers, such as

  1011010000000101

- Rather than writing programs in machine language, programmers use *programming languages*.

Some Well-Known Programming Languages



Procedural and Object-Oriented Programming

- Procedural programming: focus is on the process. Procedures/functions are written to process data.
- Object-Oriented programming: focus is on objects, which contain data and the means to manipulate the data. Messages sent to objects to perform operations.

What is a Program?

- Common elements in programming languages:
  - Key Words
  - Programmer-Defined Identifiers
  - Operators
  - Punctuation
  - Syntax

- Computer Program – a finite set of well-defined instructions that describe a task to be carried out by a computer

- Programming Language – is an artificial language used to control the computer

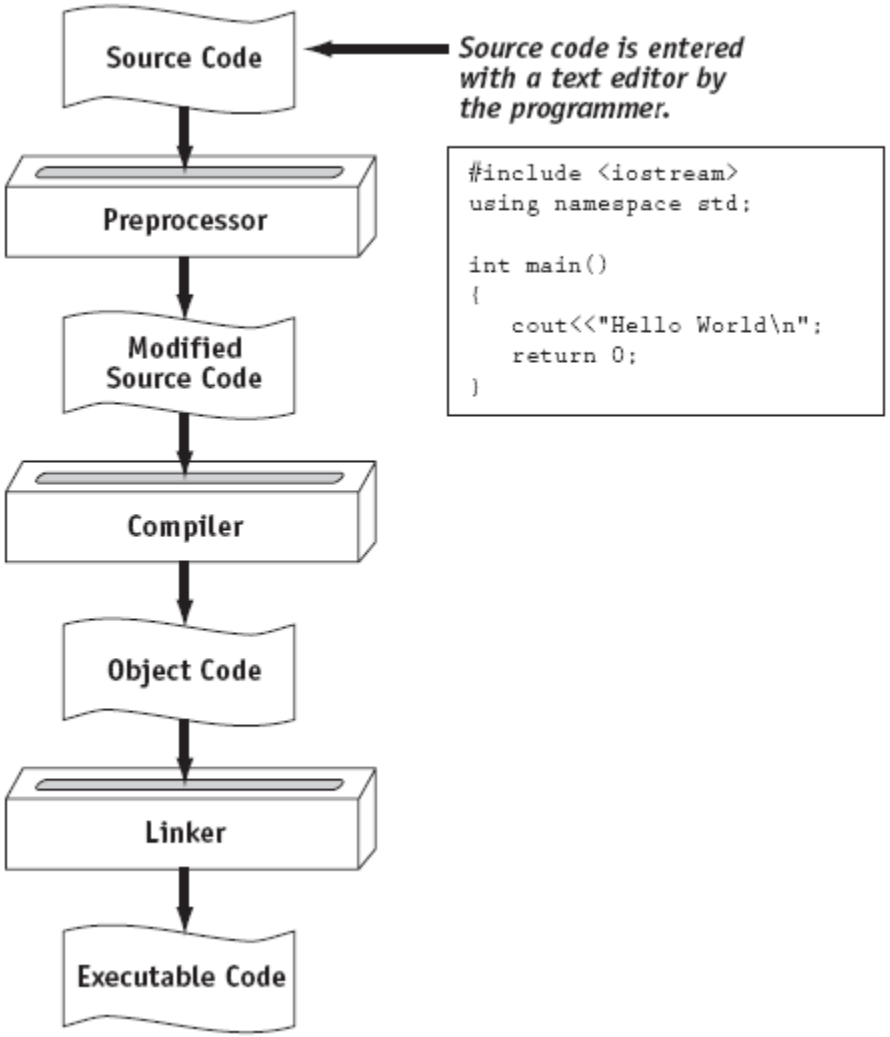
- Syntax of a Programming Language – a set of grammar rules that must be obeyed precisely

First Code, Then Run

a) Create file containing the program with a text editor.
b) Run preprocessor to convert source file directives to source code program statements.
c) Run compiler to convert source program into machine instructions.
d) Run linker to connect hardware-specific code to machine instructions, producing an executable file.

- Steps b–d are often performed by a single command or button click.
- Errors detected at any step will prevent execution of following steps.

From a High-Level Program to an Executable File



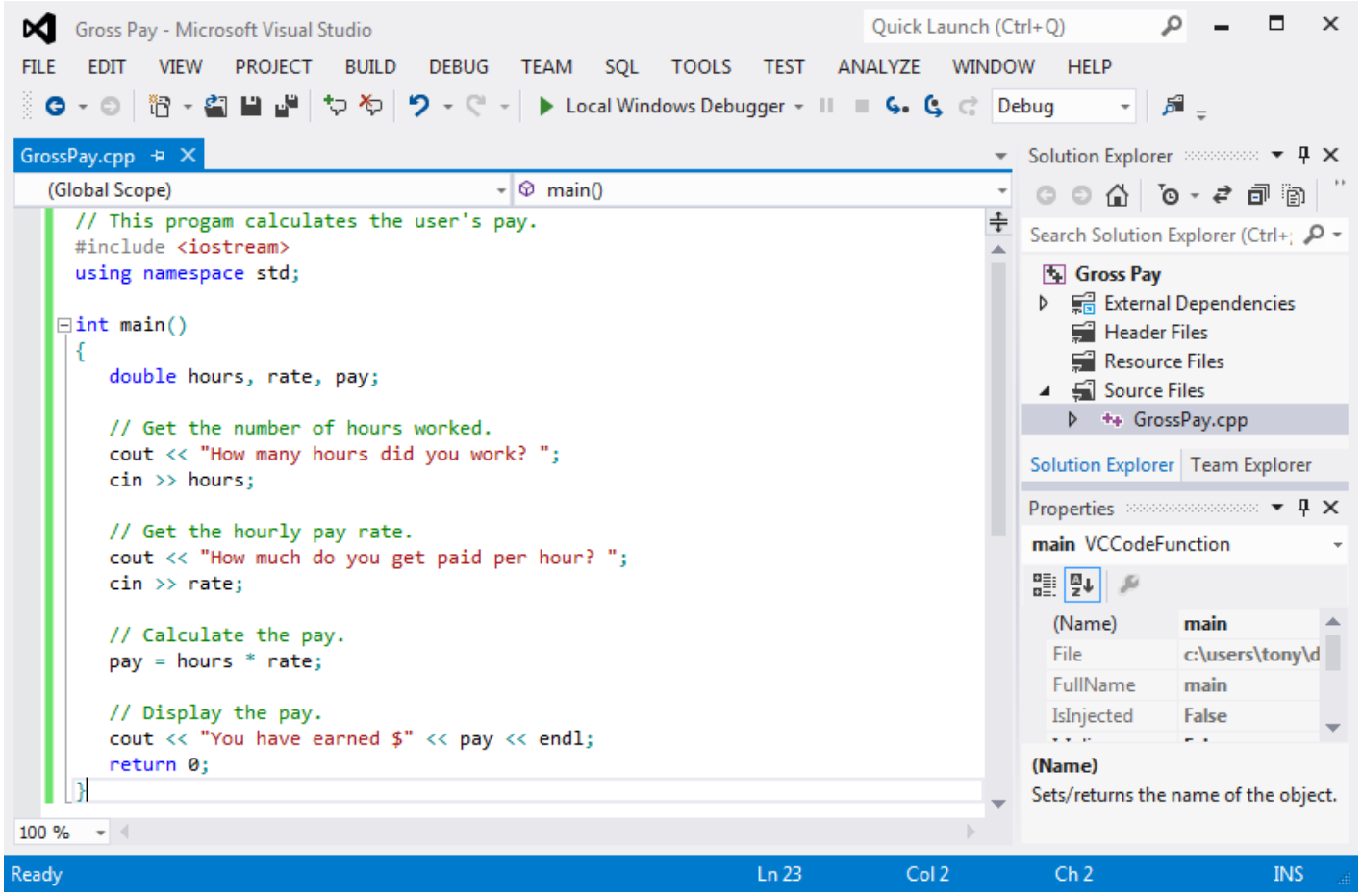
Errors

- Syntax Error – when the program is grammatically incorrect, such as missing ';'
- Logical Error – when the program does something else than it is supposed to do, using correct syntax; for instance calculate the average of three scores by dividing their sum by 2 instead of 3.
- Linking Error – when the linker fails to create the executable (some object modules are not found, etc.)
- Runtime Error – an error that occurs while the program is running; it can be caused by a conflict with other running programs, bad memory management, etc. For instance let's consider a program that divides two numbers, without checking the divisor: the program crashes when the divisor is 0.

Integrated Development Environments (IDEs)

- An integrated development environment, or IDE, combine all the tools needed to write, compile, and debug a program into a single software application.
- Examples are Microsoft Visual C++, Turbo C++ Explorer, CodeWarrior, etc.

Integrated Development Environments (IDEs)



Input, Processing, and Output

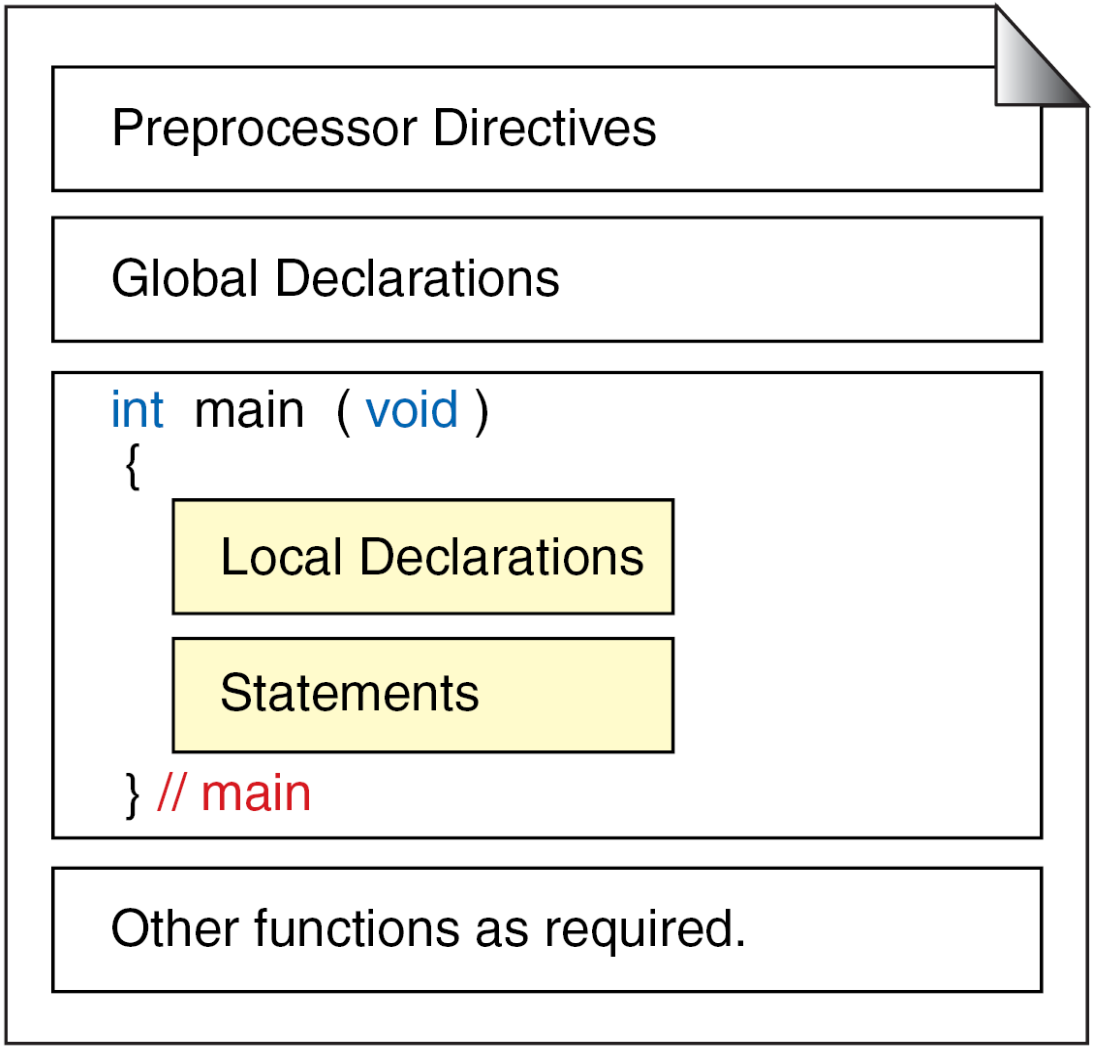
Three steps that a program typically performs:

1) **Gather input data:**
   - from keyboard
   - from files on disk drives
2) **Process the input data**
3) **Display the results as output:**
   - send it to the screen
   - write to a file

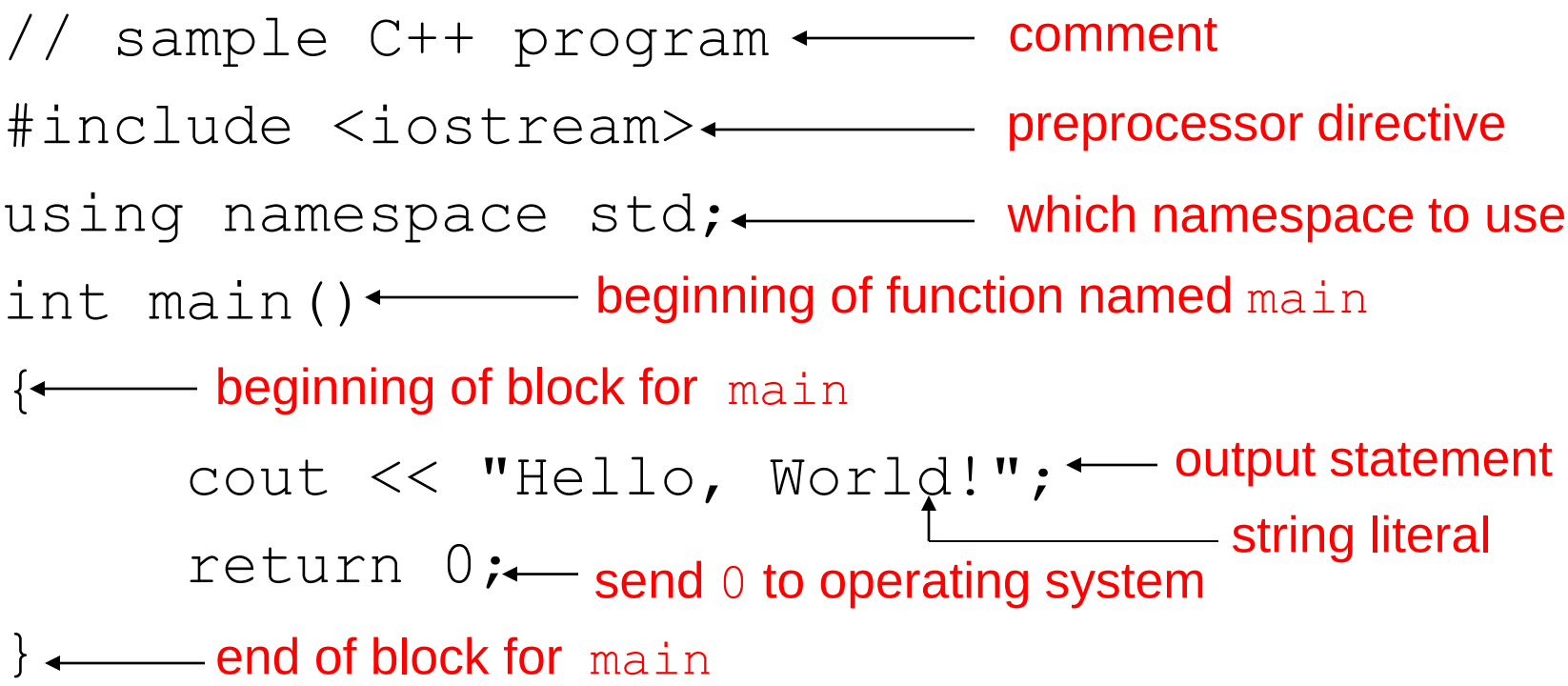
Hello, World!

```
// sample C++ program
#include <iostream>
using namespace std;
int main()
{
     cout << "Hello, World!";
     return 0;
}
```

Structure of a C++ Program



Our first program - demystified



The basic component of a C++ program is a function (also called method).

Every C++ program is made of one or more functions.

One and only one of the functions in a C++ program must be called main.

Every C++ function is made of declarations and statements and returns a value.

Other components of a C++ program

Comments Directives Declarations Data Types
Identifiers – Reserved or User Defined
Variables and Literals Operators
CLASSES and OBJECTS

Document Your Code

- Documentation is very important for:
  - Program structure → pseudocode
  - Program readability
  - Generating help information

  - Most Importantly → YOUR GRADE

Types of Comments

- Single Line Comments

```
// This is a line comment – anything after the double slashes
// on a single line is a comment
```

- Partial Line Comments

```
int myVar = 0;          // This is a line comment
```

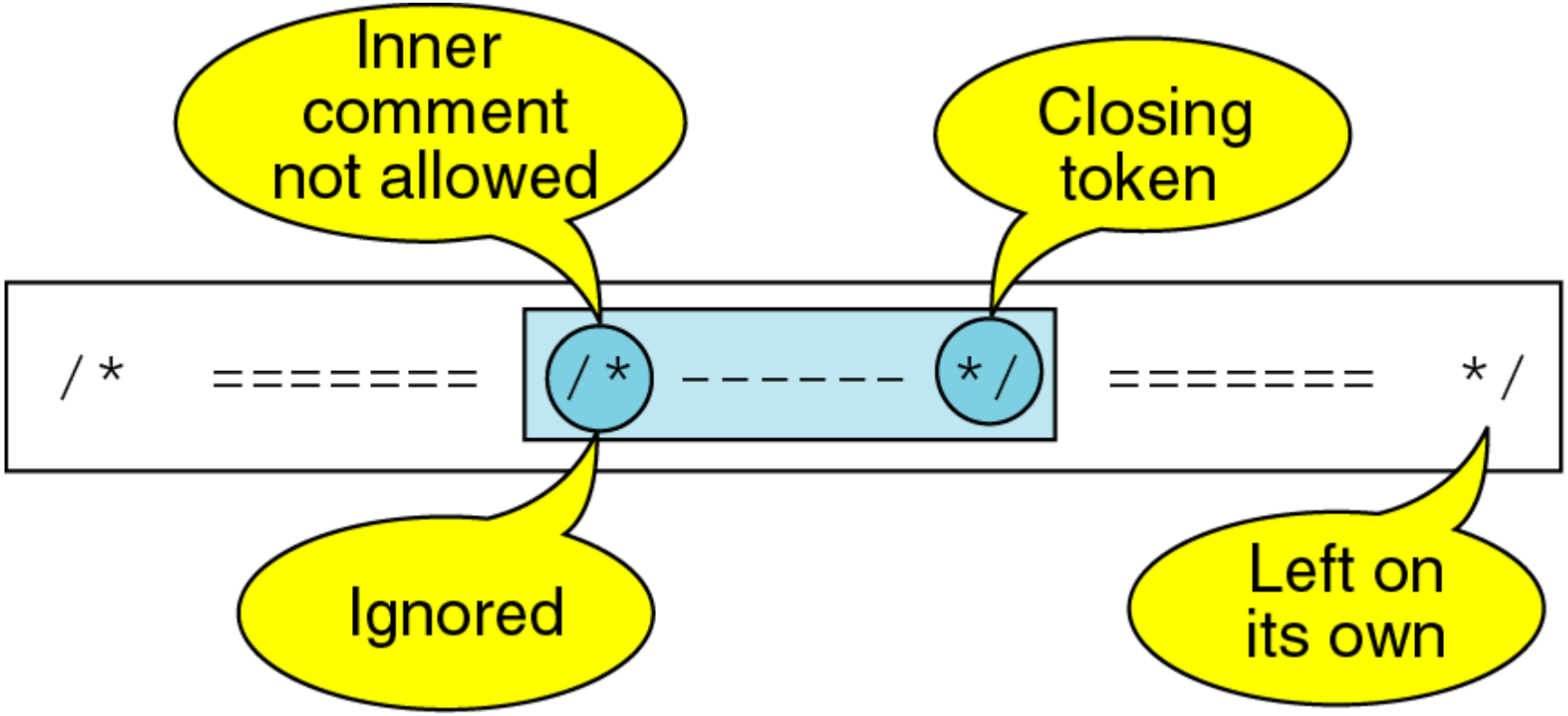
- Block Comments

```
/*
 * This is a multi-line or blockcomment – anything between the starting
 * “slash-star” and ending “star-slash” is considered to be a comment, no
 * matter how many lines it spans
 */
```

Sample of General Program Documentation

```
1  /* A sample of program documentation. Each program
2     starts with a general description of the program.
3     Often, this description can be taken from the
4     requirements specification given to the programmer.
5       Written by: original author
6       Date:       Date first released to production
7     Change History:
8       <date> Included in this documentation is a short
9              description of each change.
10 */
```

Nested Block Comments Are Invalid



The `#include` Directive

- Pre-processor directive to insert the contents of another file into the program
- `#include` lines ignored by compiler
- NO semicolon at the end of `#include` line

The `main` function

- Every program has to have a `main()`
- There can be only one `main()` as it defines the point from where a program starts executing
- The `main()` function returns control to the OS once it is finished executing – last statement in the code.
- The return-type of `main()` is `int` in C++ → the program exits to the OS with a 0 if successful or non-zero (usually -1) if there is an error.

The `cout` object

- Base C++ object – to send output to the screen.
- Part of the `iostream` library – included in the first line of the program, otherwise there will be no interaction with the program.
- The stream insertion operator `<<` is used to send the output to the `cout` stream.

HelloWorld!

```
// sample C++ program
#include <iostream>
using namespace std;
int main()
{
     cout << "Hello" << "World!";
     return 0;
}
```

Hello World!

```
// sample C++ program
#include <iostream>
using namespace std;
int main()
{
     cout << "Hello" << endl << "World!";
     return 0;
}
```


Lowercase ENDL

No quotes

Hello
World!

```
// sample C++ program
#include <iostream>
using namespace std;
int main()
{
      cout << "Hello\n" << "World!";
      return 0;
}
```


The \n is INSIDE the quotes.

(Also called ‘escape sequence’)

Special Characters

Character	Name	Meaning
//	Double slash	Beginning of a comment
#	Pound sign	Beginning of preprocessor directive
< >	Open/close brackets	Enclose filename in #include
( )	Open/close parentheses	Used when naming a function
{ }	Open/close brace	Encloses a group of statements
" "	Open/close quotation marks	Encloses string of characters
;	Semicolon	End of a programming statement