



Problem Solving / Program Logic

Queens College

CSD 1133 – FSDM -2023S



1.Introduction to Computers and Programming

Topics:

- Introduction
- Hardware
- How Computers Store Data
- How a Program Works
- Types Of Software

1.Introduction to Computers and Programming

Introduction :

Think about some of the different ways that people use computers. In school, students use computers for tasks such as writing papers, searching for articles, sending email, and participating in online classes. At work, people use computers to analyze data, make presentations, conduct business transactions, communicate with customers and coworkers, control machines in manufacturing facilities, and do many other things. At home, people use computers for tasks such as paying bills, shopping online, communicating with friends and family, and playing computer games. And don't forget that cell phones, iPods®, smart phones, car navigation systems, and many other devices are computers too. The uses of computers are almost limitless in our everyday lives.

Computers can do wide variety of things **because they can be programmed**. This means that computers are not designed to do just one job, but to do any job that their programs tell them to do. A program is a set of instructions that a computer follows to perform a task

1.Introduction to Computers and Programming

Hardware and software :

The term hardware refers to all of the physical devices, or components, that a computer is made of. A computer is not one single device, but a system of devices that all work together.

A typical computer system consists of the following major components:

- 1.The central processing unit (CPU)
- 2.Primary Storage devices - Main memory
- 3.Secondary storage devices – Hard disk drive
- 4.Input devices
- 5.Output devices

Software :

If a computer is to function, software is not optional. Everything that a computer does, from the time you turn the power switch on until you shut the system down, is under the control of software. There are two general categories of software: system software and application software.

1.Introduction to Computers and Programming

System Software :

The programs that control and manage the basic operations of a computer are generally referred to as system software. System software typically includes the following types of programs:

Operating Systems An operating system is the most fundamental set of programs on a computer. The operating system controls the internal operations of the computer's hardware, manages all of the devices connected to the computer, allows data to be saved to and retrieved from storage devices, and allows other programs to run on the computer

Utility Programs A utility program performs a specialized task that enhances the computer's operation or safeguards data. Examples of utility programs are virus scanners, file compression programs, and data backup programs.

Software Development Tools Software development tools are the programs that programmers use to create, modify, and test software. Assemblers, compilers, and interpreters are examples of programs that fall into this category.

1.Introduction to Computers and Programming

Application Software :

Programs that make a computer useful for everyday tasks are known as application software. These are the programs that people normally spend most of their time running on their computers

Examples : Microsoft Word, a word processing program, and PowerPoint, a presentation program. spreadsheet programs, email programs, web browsers, and game programs

Checkpoint :

List the five major components of a computer system.

What part of the computer serves as a work area to store a program and its data while the program is running?

What part of the computer holds data for long periods of time, even when there is no power to the computer?

What fundamental set of programs control the internal operations of the computer's hardware?

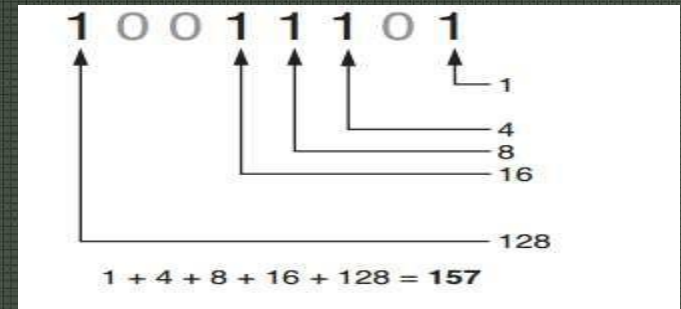
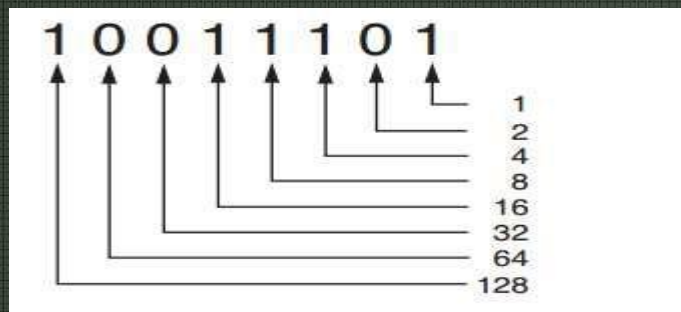
1.Introduction to Computers and Programming

How Computer Store Data?

A computer's memory is divided into tiny storage locations known as bytes. Each byte is divided into eight smaller storage locations known as bits. The term bit stands for binary digit. Computer scientists usually think of bits as tiny switches that can be either on or off.

Storing Numbers

A bit can be used in a very limited way to represent numbers. Depending on whether the bit is turned on or off, it can represent one of two different values. In computer systems, a bit that is turned off represents the number 0 and a bit that is turned on represents the number 1. This corresponds perfectly to the binary numbering system



1.Introduction to Computers and Programming

Storing Characters

Any piece of data that is stored in a computer's memory must be stored as a binary number. That includes characters, such as letters and punctuation marks. When a character is stored in memory, it is first converted to a numeric code.

The numeric code is then stored in memory as a binary number.

Over the years, different coding schemes have been developed to represent characters in computer memory. Historically, the most important of these coding schemes is ASCII, which stands for the American Standard Code for Information Interchange. ASCII is a set of 128 numeric codes that represent the English letters, various punctuation marks, and other characters. For example, the ASCII code for the uppercase letter A is 65

The ASCII character set was developed in the early 1960s, and was eventually adopted by most all computer manufacturers. ASCII is limited, however, because it defines codes for only 128 characters. **To remedy this, the Unicode character set was developed in the early 1990s.** Unicode is an extensive encoding scheme that is compatible with ASCII, and can also represent the characters of many of the world's languages. Today, Unicode is quickly becoming the standard character set used in the computer industry.

1.Introduction to Computers and Programming

How a Program Works ?

CPU is the most important component in a computer because it is the part of the computer that runs programs. Sometimes the CPU is called the “computer’s brain,” and is described as being “smart.” Although these are common metaphor, you should understand that the CPU is not a brain, and it is not smart.

CPU performs simple operations on pieces of data. The CPU does nothing on its own, however. It has to be told what to do, and that’s the purpose of a program.

A program is nothing more than a list of instructions that cause the CPU to perform operations.

Each instruction in a program is a command that tells the CPU to perform a specific operation. Here’s an example of an instruction that might appear in a program:

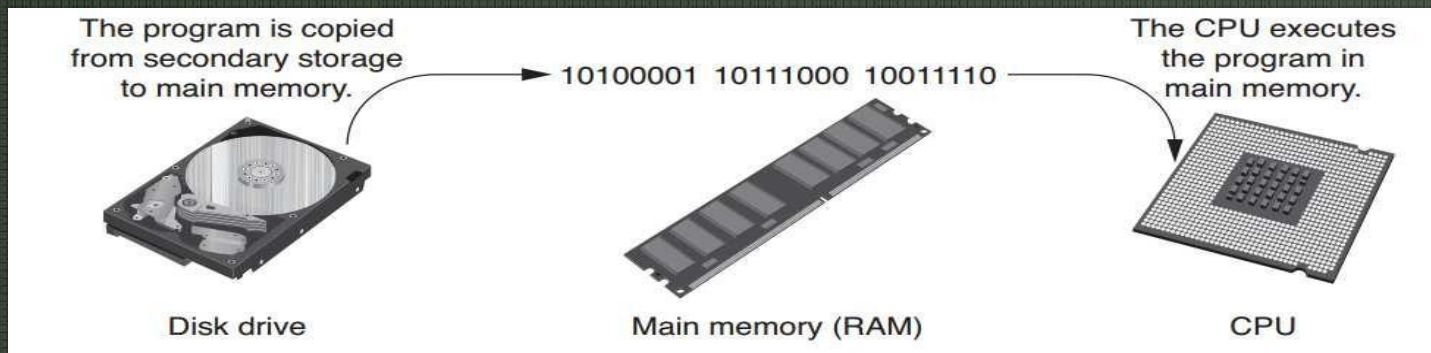
10110000

To you and me, this is only a series of 0s and 1s. To a CPU, however, this is an instruction to perform an operation¹. It is written in 0s and 1s because CPUs only understand instructions that are written in **machine language**, and machine language instructions are always written in binary.

1.Introduction to Computers and Programming

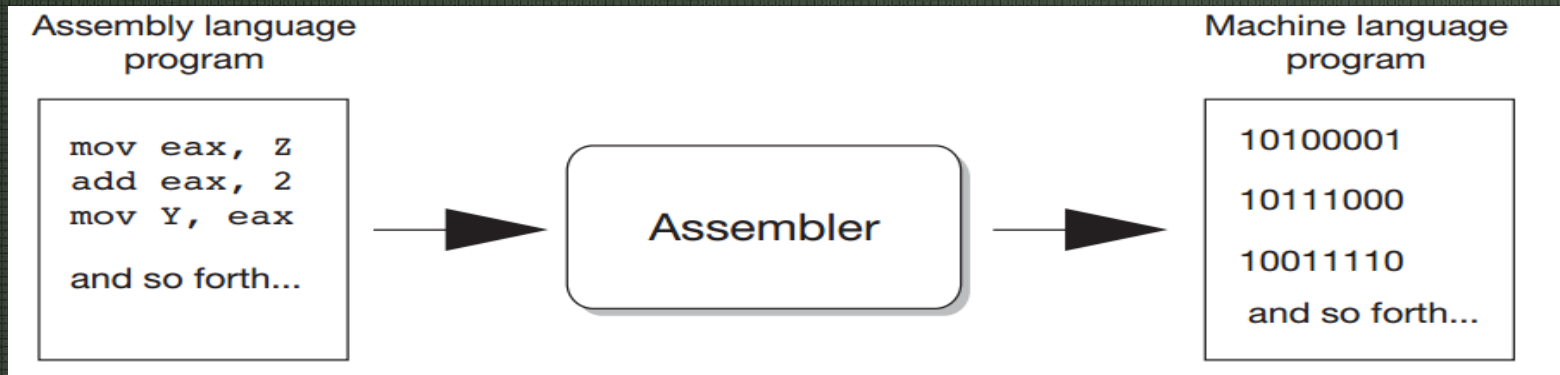
Because the operations CPU can perform are so basic in nature, a meaningful task can be accomplished only if the CPU performs many operations. It is not unusual for a program to contain thousands, or even a million or more machine language instructions.

Programs are usually stored on a secondary storage device such as a disk drive. When you install a program on your computer, the program is typically copied to your computer's disk drive from a CD-ROM, or perhaps downloaded from a Web site. Although a program can be stored on a secondary storage device such as a disk drive, it has to be copied into main memory, or RAM, each time the CPU executes it



1.Introduction to Computers and Programming

Although a computer's CPU only understands machine language, it is impractical for people to write programs in machine language. For this reason, assembly language was created in the early days of computing as an alternative to machine language. Instead of using binary numbers for instructions, assembly language uses short words that are known as mnemonics.



Assembly language also requires that you write a large number of instructions for even the simplest program. Because assembly language is so close in nature to machine language, it is referred to as a low-level language.

In the 1950s, a new generation of programming languages known as **high-level languages** began to appear. A high-level language allows you to create powerful and complex programs without knowing how the CPU works, and without writing

1.Introduction to Computers and Programming

large numbers of low-level instructions. In addition, most high-level languages use words that are easy to understand.

Programming languages :

Ada - Ada was created in the 1970s, primarily for applications used by the U.S. Department of Defense. The language is named in honor of Countess Ada Lovelace, an influential and historical figure in the field of computing.

BASIC - Beginners All-purpose Symbolic Instruction Code is a general-purpose language that was originally designed in the early 1960s to be simple enough for beginners to learn. Today, there are many different versions of BASIC.

FORTRAN - FORmula TRANslator was the first high-level programming language. It was designed in the 1950s for performing complex mathematical calculations.

COBOL Common Business-Oriented Language was created in the 1950s, and was designed for business applications.

Pascal Pascal was created in 1970, and was originally designed for teaching programming. The language was named in honor of the mathematician, physicist, and philosopher Blaise Pascal.

1.Introduction to Computers and Programming

C# Pronounced “c sharp.” This language was created by Microsoft around the year 2000 for developing applications based on the Microsoft .NET platform.

Java Java was created by Sun Microsystems in the early 1990s. It can be used to develop programs that run on a single computer or over the Internet from a Web server.

JavaScript™ JavaScript, created in the 1990s, can be used in Web pages. Despite its name, JavaScript is not related to Java.

Python Python is a general-purpose language created in the early 1990s. It has become popular in business and academic applications.

Ruby Ruby is a general-purpose language that was created in the 1990s. It is increasingly becoming a popular language for programs that run on Web servers.

Visual Basic Visual Basic (commonly known as VB) is a Microsoft programming language and software development environment that allows programmers to create Windows®-based applications quickly. VB was originally created in the early 1990s

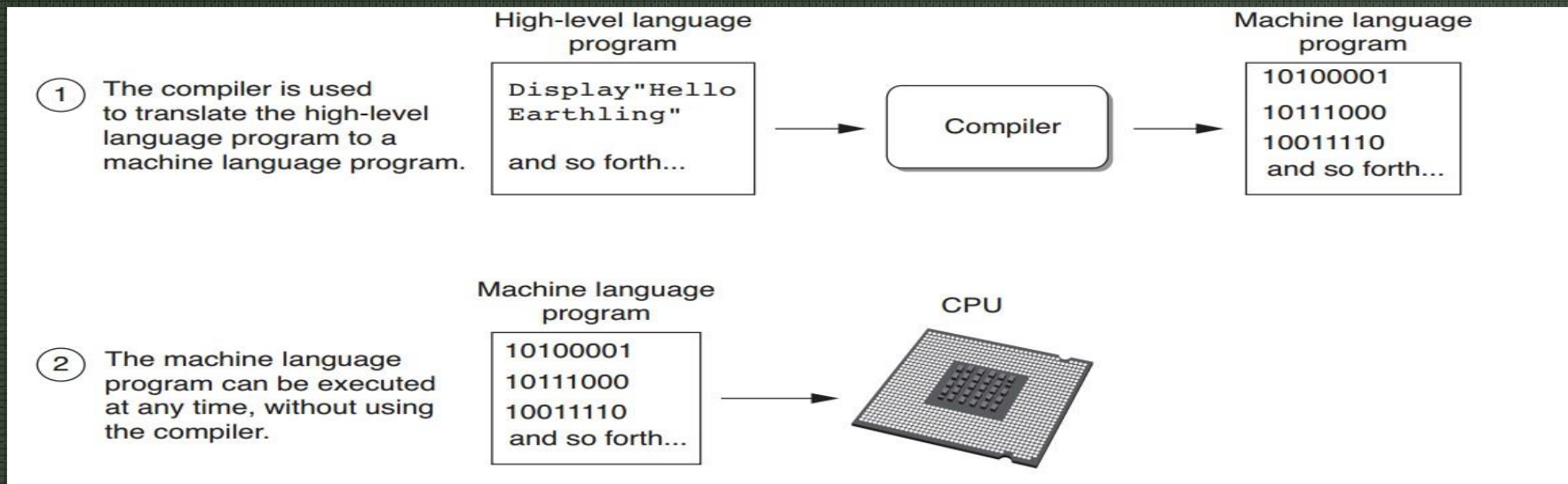
1.Introduction to Computers and Programming

Compilers and Interpreters

Because the CPU understands only machine language instructions, programs that are written in a high-level language must be translated into machine language.

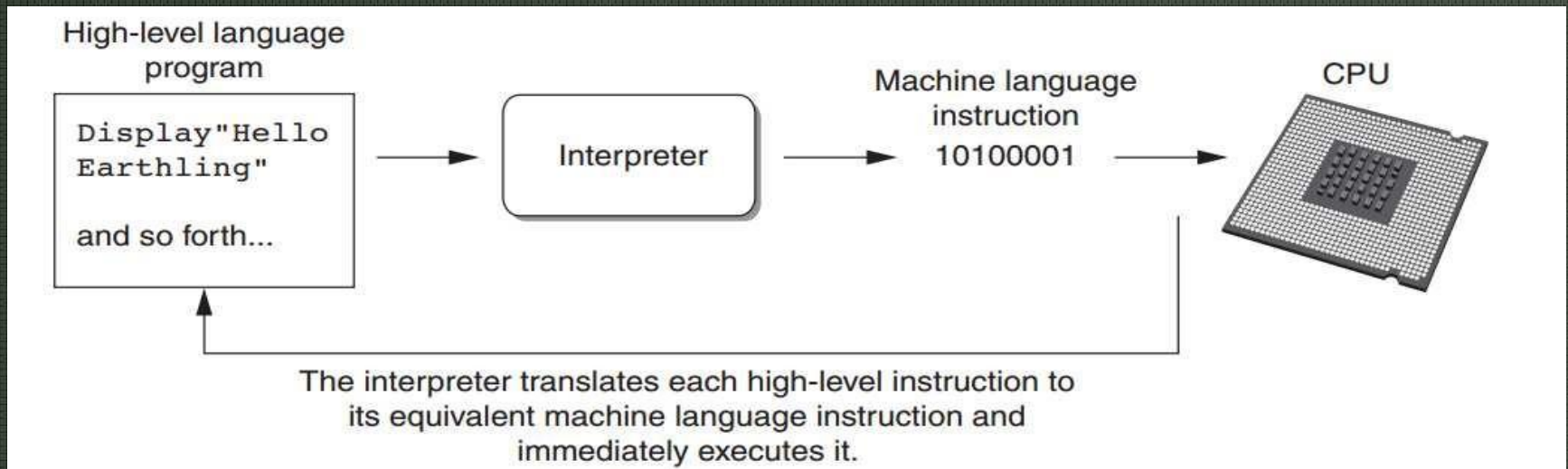
Once a program has been written in a high-level language, the programmer will use a **compiler** or an **interpreter** to make the translation

A **compiler** is a program that translates a high-level language program into a separate machine language program. The machine language program can then be executed any time it is needed.



1.Introduction to Computers and Programming

An *interpreter* is a program that both translates and executes the instructions in a highlevel language program. As the interpreter reads each individual instruction in the program, it converts it to a machine language instruction and then immediately executes it. This process repeats for every instruction in the program. Because interpreters combine translation and execution, they typically do not create separate machine language programs.



1.Introduction to Computers and Programming

Types of software :

If a computer is to function, software is not optional. Everything that a computer does, from the time you turn the power switch on until you shut the system down, is under the control of software. There are two general categories of software: **system software and application software**. Most computer programs clearly fit into one of these two categories.

System Software:

The programs that control and manage the basic operations of a computer are generally referred to as system software. System software typically includes the following types of programs:

Operating Systems. An operating system is the most fundamental set of programs on a computer. The operating system controls the internal operations of the computer's hardware, manages all of the devices connected to the computer, allows data to be saved to and retrieved from storage devices, and allows other programs

1.Introduction to Computers and Programming

Operating Systems. An operating system is the most fundamental set of programs on a computer. The operating system controls the internal operations of the computer's hardware, manages all of the devices connected to the computer, allows data to be saved to and retrieved from storage devices, and allows other programs to run on the computer

Utility Programs. A utility program performs a specialized task that enhances the computer's operation or safeguards data. Examples of utility programs are virus scanners, file compression programs, and data backup programs.

Software Development Tools. Software development tools are the programs that programmers use to create, modify, and test software. Assemblers, compilers , and interpreters are examples of programs that fall into this category

1.Introduction to Computers and Programming

Application Software:

Programs that make a computer useful for everyday tasks are known as application software. These are the programs that people normally spend most of their time running on their computers. Some of the commonly used application software programs are Microsoft Word, a word processing program, and Microsoft PowerPoint, a presentation program. Some other examples of application software are spreadsheet programs, email programs, Web browsers, and game programs



1.Introduction to Computers and Programming

Integrated Development Environments(IDE)

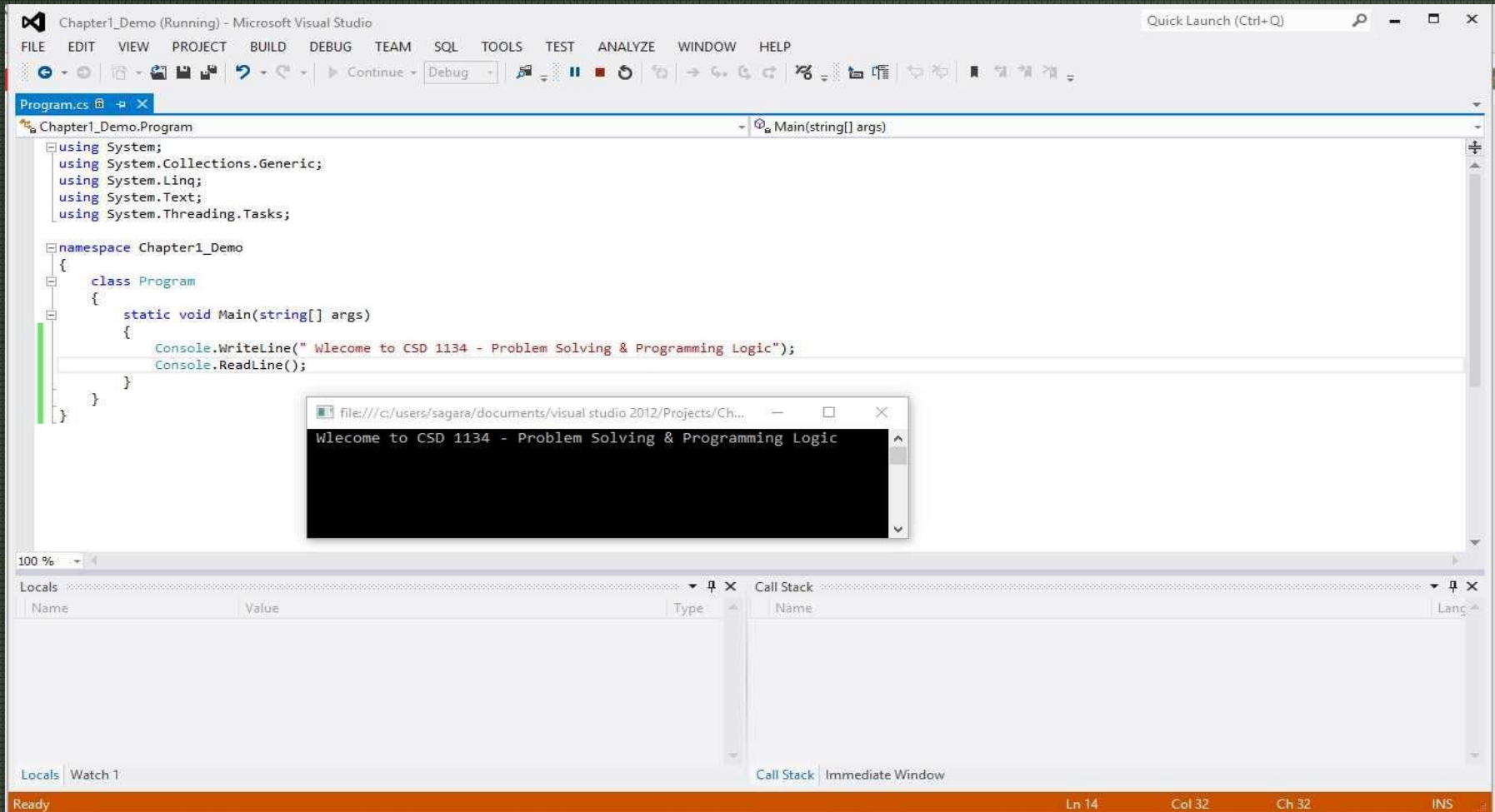
Although you can use a simple text editor such as Notepad (which is part of the Windows operating system) to write a program, most programmers use specialized software packages called integrated development environments or IDEs. Most IDEs combine the following programs into one software package:

- A text editor that has specialized features for writing statements in a high-level programming language
- A compiler or interpreter
- Useful tools for testing programs and locating errors

Microsoft Visual Studio , Eclipse™, NetBeans, and Dev-C++are a few other popular IDEs.

1.Introduction to Computers and Programming

Following Figure shows a screen from Microsoft Visual Studio, a popular IDE for developing programs in the C++, Visual Basic, and C# languages.



1.Introduction to Computers and Programming

Popular Programming Languages – 2017 August Vs 2021 January

Popularity of Programming Language in August 2017...

Popularity of Programming Language

Rank	Language	Share	Trend
1	Java	22.7 %	-0.7 %
2	Python	16.3 %	+3.7 %
3	PHP	8.9 %	-1.1 %
4	C#	8.3 %	-0.5 %
5	Javascript	8.0 %	+0.5 %
6	C++	6.6 %	-0.2 %
7	C	6.4 %	-0.7 %
8	R	3.6 %	+0.4 %
9	Objective-C	3.6 %	-1.2 %
10	Swift	2.8 %	-0.3 %
11	Matlab	2.3 %	-0.2 %
12	Ruby	1.8 %	-0.5 %
13	VBA	1.5 %	+0.0 %
14	Visual Basic	1.4 %	-0.3 %
15	TypeScript	1.3 %	+0.4 %
16	Scala	1.2 %	+0.1 %
17	Perl	0.8 %	-0.3 %
18	Go	0.6 %	+0.2 %

Worldwide, Jan 2021 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	30.44 %	+1.2 %
2		Java	16.76 %	-2.0 %
3		JavaScript	8.44 %	+0.3 %
4		C#	6.53 %	-0.7 %
5	↑	C/C++	6.33 %	+0.3 %
6	↓	PHP	6.05 %	-0.2 %
7		R	3.87 %	+0.1 %
8		Objective-C	3.71 %	+1.2 %
9		Swift	2.14 %	-0.3 %
10		TypeScript	1.78 %	-0.0 %

1.Introduction to Computers and Programming

Checkpoint :

A CPU understands instructions that are written only in what language?

A program has to be copied into what type of memory each time the CPU executes it?

What is assembly language?

What do you call a program that translates a high-level language program into a separate machine language program?

What do you call a program that both translates and executes the instructions in a high-level language program?

What type of mistake is usually caused by a misspelled key word, a missing punctuation character, or the incorrect use of an operator?