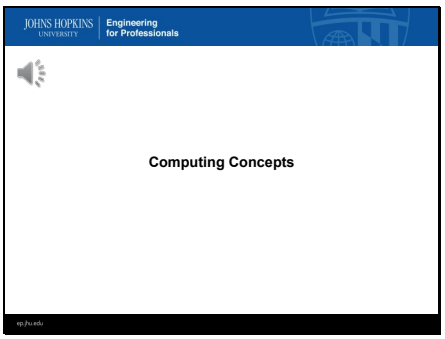
1



In this lesson you will learn some basic things about computers, computer components, and related terminology that will help you better understand what goes on behind the scenes when you develop a computer program.

2

A **computer** is an electronic device capable of performing computations and making logical decisions at speeds much, much greater than humans can.

Computers work at speeds millions, and even billions, of times faster than a human.

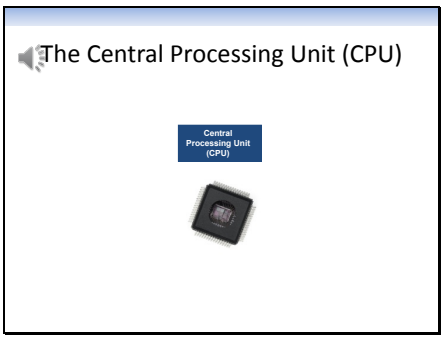
There are a lot of different types of computers in existence today. You are probably taking this course using a personal computer or a laptop computer…and there are many different types of personal computers and laptops available. In fact, I'm developing this course on a very small computer called a netbook that weighs slightly more than one pound.

There are also very large computers, like this supercomputer, that take up an entire computer center.

Smart cell phones and devices like the Apple iPad are also types of computers.

There are certain things that all computers have in common, regardless of their type.

3



The heart of a computer is the Central Processing Unit or CPU. This is where the computing takes place.

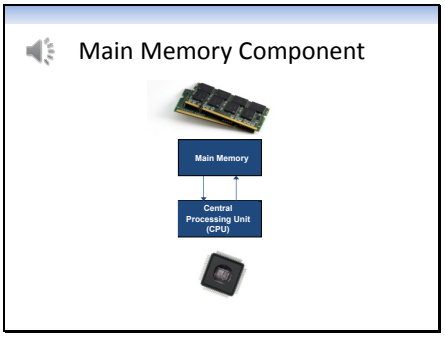
The actual computing consists of the CPU executing a set of instructions, called a **computer program**, that tell it what to do.

An instruction is loaded into the CPU, then it is executed, then another instruction is loaded and executed, and the process continues until there are no more instructions left to execute. And this happens very, very quickly.

CPUs for personal computers and laptops are very small…about one inch square in size. The programs that run on a computer are commonly referred to as **software**.

A word processing program and an Internet browser program, like Internet Explorer are examples of software, and the programs you will write as part of this course are also software.

4



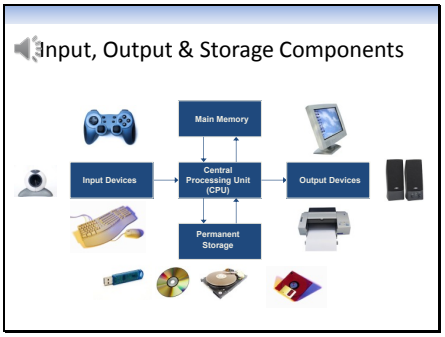
A computer also has a memory component, sometimes called the main memory.

The memory component stores information like programming instructions and data that the computer needs access to when it performs its computations.

This type of memory component is also called **random access memory**, or **RAM**, and is designed so that the computer can read and write information very quickly.

The main memory only stores information while a program is running or while the computer's power is turned on. There are other types of memory that computers use, but we do not discuss them in this course.

5



Another common computer component is **permanent storage**. Permanent storage devices are used to store information that needs to be saved when the computer's power is turned off or programs stop running. The most common type of permanent storage devices are disk drive devices, usually called hard disks, that may be built into a computer or may connect to it externally.

Portable storage media such as flash drives, compact disks and DVDs, and floppy disks are typically used to store information when computers are turned off and can be moved from computer to computer.

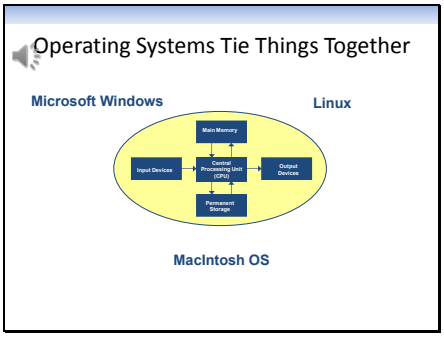
A computer's input devices are used to input information.

The most common input devices are a keyboard and a mouse. Devices such as a joystick, and a webcam are other examples of input devices.

Computers can output information from computer programs to many different types of output devices like a video display, a printer, and speakers.

Hardware is a term that refers to the parts of a computer that you can see and touch. All of the computer components we've mentioned so far are examples of hardware.

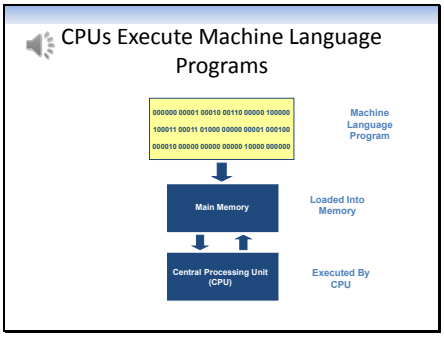
6



All of these computer components are coordinated by a computer's **operating system**, which is a special type of computer program that usually comes already installed when you purchase a computer.

Popular operating systems for today's personal and laptop computers are Windows, Macintosh OS, and Linux.

7



As mentioned earlier in this lesson, the main job of a computer's CPU is to execute a set of instructions, called a program. Computer CPUs are designed to execute instructions that are written in a very simple low-level language called **machine language**.

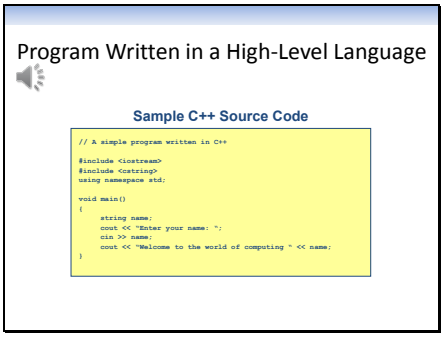
Machine language programs are written using binary numbers…consisting only of zeros and ones…and are very difficult to understand…at least by humans. It is certainly not obvious what this sample machine code program is trying to accomplish. Not to worry though…you won't be writing any machine language programs in this course.

For a computer to run a machine language program, the program, which may be stored on some storage

device like a hard disk, has to be loaded into the computer's memory. Then, the program is started and the CPU executes each instruction until it reaches the end of the program.

Each different family of CPUs has its own machine language and only knows how to execute a program written in that language. So, this program would have to be re-written for each different type of processor that a programmer wants to run it on.

8



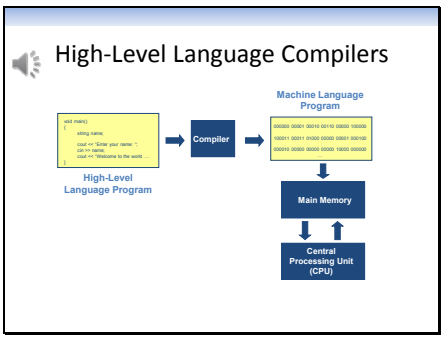
Most people who write computer programs today don't write programs in machine language, but write them in a **high-level language**, like Java or C or C++. These higher-level languages make it easier, and faster, to write very complex programs.

This is an example of a very simple program written in the C++ language. The program simply prompts the user to enter their name, then displays the message "welcome to the world of computing" followed by the person's name.

Even though you may not understand all of the programming instructions…you'd probably agree that it's a lot easier to begin to understand than the earlier machine language program…because it's more English-like…than a bunch of zeros and ones.

By the way, the set of high-level language instructions that make up a program are commonly referred to as the program's **source code**.

9



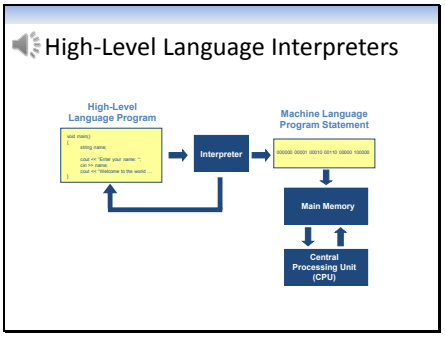
Now here's the thing. Computers can't execute programs written in high-level languages unless they are first translated into machine language. The most common way to translate a program written in a high-level language into machine language is by using a **compiler**.

A compiler, which is itself a computer program, reads a program written in a high-level language and translates it into an executable machine language program that a particular type of computer can understand.

Once the translation is done, the machine language program can be executed, or run, many times, but it can only be run on one type of computer.

To run the program on a different type of computer, the high-level language program would have to be re-translated using a compiler designed for the different type of computer. Computer programming languages like C and C++ use this type of compiler.

10



An alternative to using a compiler to translate a program written in a high-level language is to use something called an interpreter.

There are different types of interpreters… and I've illustrated a very simple one here so that you can compare it to the compiler approach.

In this case, instead of translating an entire computer program all at once into machine language, this interpreter translates instructions one at a time.

For this particular interpreter approach, the computer program would run more slowly than in the compiled approach, because the interpreter is essentially feeding the CPU one machine language statement at a time.

Just like in the compiler approach, in order to run a program on different types of computers using this

type of interpreter would require that we have an interpreter designed for each different type of computer.

As we'll see in the next lesson, Java is designed to use a combination of the compilation and interpretation approaches.