

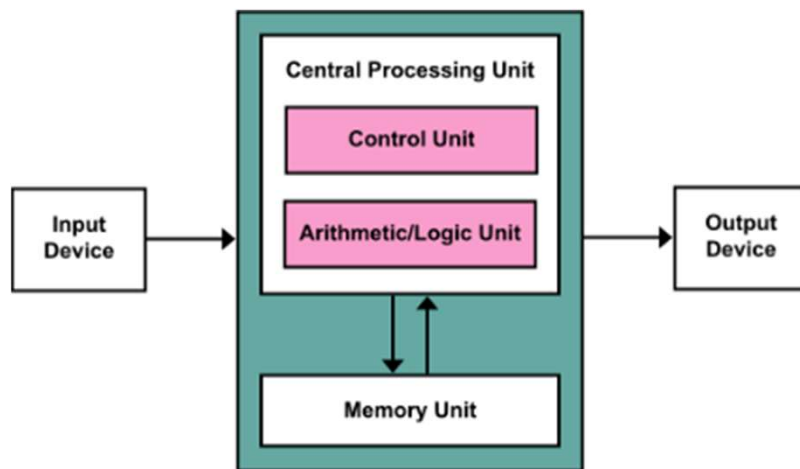
Intro To Computers and Python

Lec-2

H/W and S/W

- Computers can perform calculations and make logical decisions phenomenally faster than human beings can.
- Computers process data under the control of sequences of instructions called computer programs (or simply programs).
- The person writing those programs is called a programmer.
- Every year or two, the capacities of computers have approximately doubled inexpensively. This remarkable trend often is called Moore's Law. *
- Law and related observations apply especially to
 - the amount of memory that computers have for programs,
 - the amount of secondary storage (such as solid-state drive storage) they have to hold programs and data over longer periods of time, and
 - their processor speeds—the speeds at which they execute their programs (that is, do their work).

H/W and S/W



Input Unit

This “receiving” section obtains information (data and computer programs) from input devices and places it at the disposal of the other units for processing. Most user input is entered into computers through keyboards, touch screens and mouse devices.

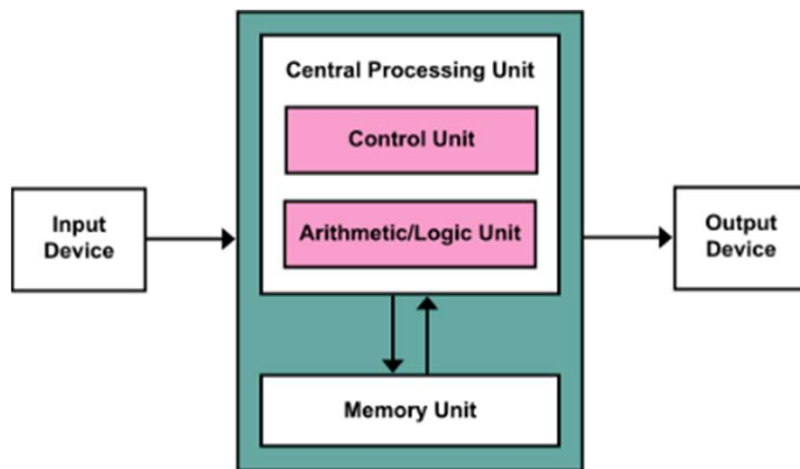
Output Unit

This “shipping” section takes information the computer has processed and places it on various output devices to make it available for use outside the computer.

Memory Unit

Memory Unit This rapid-access, relatively low-capacity “warehouse” section retains information that has been entered through the input unit, making it immediately available for processing when needed. The memory unit also retains processed information until it can be placed on output devices by the output unit. Information in the memory unit is volatile—it’s typically lost when the computer’s power is turned off.

H/W and S/W



Arithmetic Logic Unit

This “manufacturing” section performs calculations, such as addition, subtraction, multiplication and division. It also contains the decision mechanisms that allow the computer, for example, to compare two items from the memory unit to determine whether they’re equal. In today’s systems, the ALU is part of the next logical unit, the CPU.

Control Unit

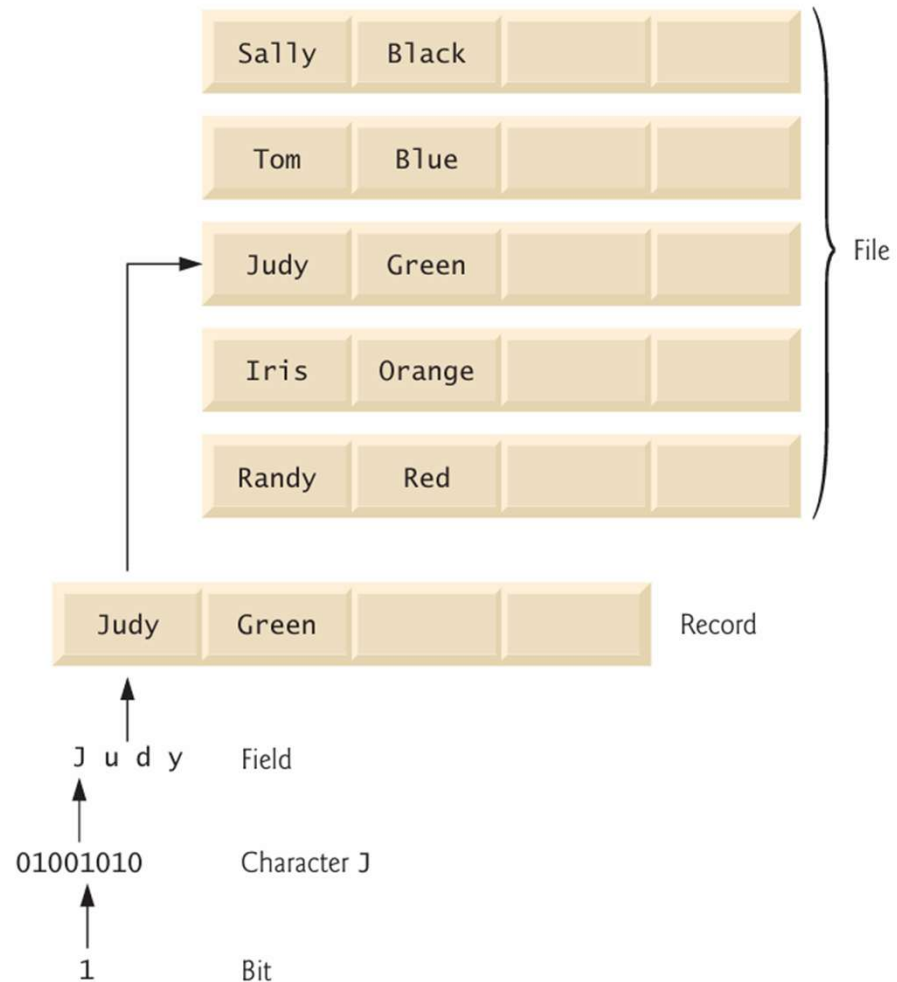
This “administrative” section coordinates and supervises the operation of the other sections. The CPU tells the input unit when information should be read into the memory unit, tells the ALU when information from the memory unit should be used in calculations and tells the output unit when to send information from the memory unit to specific output devices.

Secondary Storage Unit

This is the long-term, high-capacity “warehousing” section. Programs or data not actively being used by the other units normally are placed on secondary storage devices (e.g., your hard drive) until they’re again needed, possibly hours, days, months or even years later. Information on secondary storage devices is persistent—it’s preserved even when the computer’s power is turned off. Secondary storage information takes much longer to access than information in primary memory, but its cost per unit is much less.

Data Hierarchy

- Data items processed by computers form a data hierarchy that becomes larger and more complex in structure as we progress from the simplest data items (called “bits”) to richer ones, such as characters and fields
- What is a bit, character, field, record, file, database?



Machine/Assembly/High Level Languages

- What is a machine language? What is an assembly language? What is a high level language?

Machine Language

Any computer can directly understand only its own machine language, defined by its hardware design. Machine languages generally consist of strings of numbers (ultimately reduced to 1s and 0s) that instruct computers to perform their most elementary operations one at a time. Machine languages are machine dependent (a particular machine language can be used on only one type of computer).

Machine/Assembly/High Level Languages

Assembly Language

Programming in machine language was simply too slow and tedious for most programmers. Instead of using the strings of numbers that computers could directly understand, programmers began using English-like abbreviations to represent elementary operations. These abbreviations formed the basis of assembly languages. Translator programs called assemblers were developed to convert assembly-language programs to machine language at computer speeds.

High Level Language

Programmers still had to use numerous instructions to accomplish even the simplest tasks. To speed the programming process, high-level languages were developed in which single statements could be written to accomplish substantial tasks. A typical high-level-language program contains many statements, known as the program's source code.

Compilers and Interpreters

Compiler

Translator programs called compilers convert high-level-language source code into machine language.

Interpreter

Compiling a large high-level language program into machine language can take considerable computer time. Interpreter programs, developed to execute high-level language programs directly, avoid the delay of compilation, although they run slower than compiled programs.

Introduction to Objects

- Objects, or more precisely, the classes objects come from, are essentially reusable software components. There are date objects, time objects, audio objects, video objects, automobile objects, people objects, etc. Almost any noun can be reasonably represented as a software object in terms of attributes (e.g., name, color and size) and behaviors (e.g., calculating, moving and communicating).
- Python is an Object-Oriented Programming Language. Everything is an object.
- `x = 9 + 7j` # x is an object of type complex
- `print(x.real, x.imag)`

Operating Systems

- Operating systems are software systems that make using computers more convenient for users, application developers and system administrators.
- They provide services that allow each application to execute safely, efficiently and concurrently with other applications.
- The software that contains the core components of the operating system is called the kernel.
- Ex Windows, Linux, MacOS, AndroidOS, iOS

Python

- Python is an object-oriented scripting language that was released publicly in 1991. It was developed by Guido van Rossum of the National Research Institute for Mathematics and Computer Science in Amsterdam.
- Why popular?
- It's open source, free and widely available with a massive open-source community.
- It's easier to learn than languages like C, C++, C# and Java, enabling novices and professional developers to get up to speed quickly.
- It's easier to read than many other popular programming languages.
- It's widely used in education.¹⁵
- It enhances developer productivity with extensive standard libraries and *thousands* of third-party open-source libraries, so programmers can write code faster and perform complex tasks with minimal code. We'll say more about this in Section 1.8.
- There are massive numbers of free open-source Python applications.
- It's popular in web development (e.g., Django, Flask).
- It supports popular programming paradigms—procedural, functional, object-oriented and reflective.¹⁶ We'll begin introducing functional-style programming features in Chapter 4 and use them in subsequent chapters.
- It simplifies concurrent programming—with `asyncio` and `async/await`, you're able to write single-threaded concurrent code¹⁷, greatly simplifying the inherently complex processes of writing, debugging and maintaining that code.¹⁸
- There are lots of capabilities for enhancing Python performance.
- It's used to build anything from simple scripts to complex apps with massive numbers of users, such as Dropbox, YouTube, Reddit, Instagram and Quora.¹⁹

Questions

- (Fill-In) Every year or two, the capacities of computers have approximately doubled inexpensively. This remarkable trend often is called _____.
- (True/False) Information in the memory unit is persistent—it's preserved even when the computer's power is turned off.
- A digit that can assume one of two values) is the smallest data item in a computer is known as a _____.
- Translator programs called _____ convert assembly-language programs to machine language at computer speeds.
- _____ programs, developed to execute high-level-language programs directly, avoid the delay of compilation, although they run slower than compiled programs.
- (True/False) High-level languages allow you to write instructions that look almost like everyday English and contain commonly used mathematical notations.
- The size, shape, color and weight of an object are of the object's class.

Questions

- (True/False) Objects, or more precisely, the classes objects come from, are essentially reusable software components.
- (True/False) The Python language supports popular programming paradigms—procedural, functional, object-oriented and reflective.