Introduction to Programming in Python

What is a Computer?

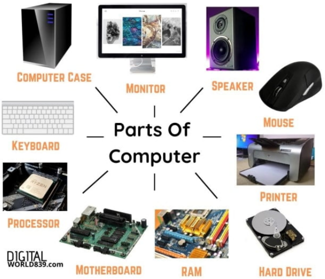
Dr. Bill Young
Department of Computer Science
University of Texas at Austin

Last updated: June 4, 2021 at 11:04

What is a Computer?

A typical computer consists of:

- a CPU
- ram memory
- a hard drive
- a monitor
- a keyboard and possibly a mouse
- one or more communication devices



CPU (Central Processing Unit

The CPU is the “brain” of a computer. It retrieves instructions from memory and executes them.

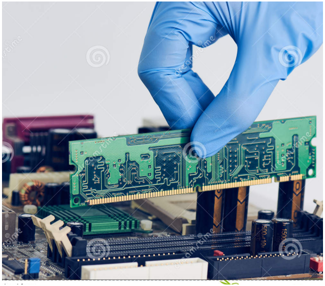
CPU speed was originally measured in megahertz (MHz): 1 MHz = 1 million cycles per second.

CPU speed has improved continuously. Intel’s Core i7 runs at 3.6 GHz, where 1 GHz = 1 billion cycles per second.

RAM Memory

RAM (random access memory) is a place to store data and programs for the CPU to execute. It is not the same thing as secondary storage (disk, flash, etc.).

- A program and its data must be copied into memory before they can execute.
- A memory unit is an ordered sequence of *bytes*, each of which holds eight *bits*.
- Memory is never empty, but its initial contents may be meaningless to your program. Writing a byte to memory overwrites what's there.

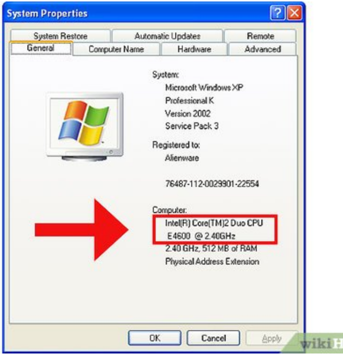


The “Fetch-Execute Cycle”

The Central Processing Unit (CPU) in a computer executes a “fetch-execute cycle” over and over.

- Fetch an instruction
- Execute the instruction
- Repeat

The speed of this cycle is defined by the speed of the processor chip.



System:
Microsoft Windows XP
Professional K
Version 2002
Service Pack 3

Registered to:
Alienware

76487-112-0029901-22554

Computer:
Intel(R) Core(TM)2 Duo CPU
E4600 @ 2.40GHz
2.40 GHz, 512 MB of RAM
Physical Address Extension

Programs and Programming Languages

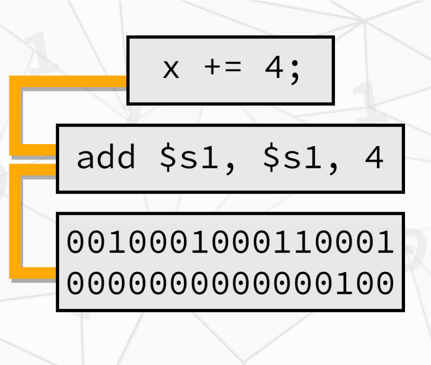
Computer programs or software are simply sets of instructions to the computer.

A programmer tells a computer what to do through programs. Without software, a computer is an empty machine.

Computers do not understand human languages, so you need to use computer languages to communicate with them. Programs are written using various different programming languages.

Machine Language

When you write a program in Python (or any other *high-level language*) it must be translated into *machine language* to run.



```
x += 4;
```

```
add $s1, $s1, 4
```

```
0010001000110001
0000000000000100
```

Machine language is a set of primitive instructions built into every computer. It is the language that that computer's CPU chip "understands."

The instructions are in the form of binary code, so you have to enter binary codes for various instructions.

Machine Language

Programming in a native machine language is a very tedious process. Moreover, the programs are highly difficult to read and modify.

```
10011101000110100000
01100011010001110110
10000010111101101110
11110110001011011000
10000010011100011011
10010011000111000000
```

High-Level Programming Languages

High-level programming languages are even more English-like and easy to learn and program.

The following is an example of Python code to compute the area of a circle with radius 5:

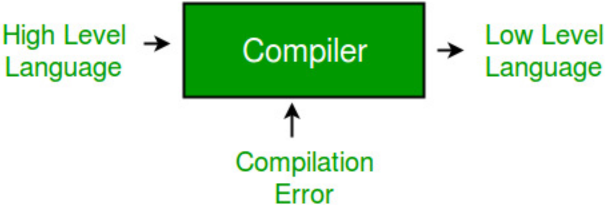
```
r = 5
area = math.pi * r ** 2
```

This statement would actually be translated into multiple machine language instructions.

Compilers vs. Interpreters

Compilers are programs that take a source file written in a high-level language, translate them, and produce an object file written in machine language.

The object file can then be executed by the computer.

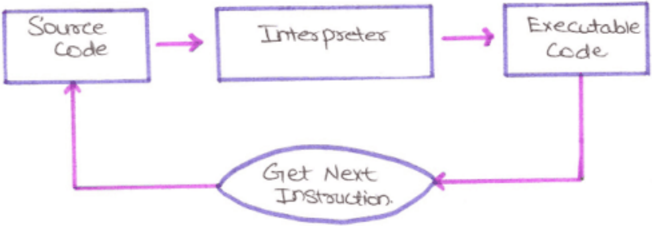


C, C++, and Fortran are typically compiled languages.

Compilers vs. Interpreters

Interpreters are programs that take individual instructions written in a high-level language, translate them, and execute them.

You can also accumulate instructions in a file and have the interpreter automatically run them one at a time.



Python, Ruby, and Swift are typically interpreted languages.

The Python Interpreter

This is an example of me running the Python interpreter:

```
> python
Python 3.6.9 (default, Jan 26 2021, 15:33:00)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more
    information.
>>> import math
>>> r = 5
>>> area = math.pi * r ** 2
>>> area
78.53981633974483
>>>
```

Your command to start python may be different. I defined the command `python` to open `python3`. Be sure you're using Python-3.