01: Introduction to Computers and Programming

Introduction to Computers and Programming

Hardware

Programming Languages

What are Programming Languages Made Of?

Variables

Input and Output

The Programming Process

Introduction to Computers and Programming

How are programs made?

This is a general discussion.

How are programs made?

- Developers, using a **programming language**, translate a problem into that language, creating **code**.
- Developers then pass that **code** into a **compiler**. This creates a file called an executable. They give that file to someone who will use the software.
- The user will then ask the **operating system** (such as Windows, Mac OSX, or Linux) to then see the code in action.

You will become the **developer** at this camp.

Important Terms

- Computer: programmable machine designed to follow instructions
- Program: instructions in computer memory that solve a problem
- Programmer: person who writes instructions (i.e. programs) to make the computer perform a task

Without a programmer, there are no programs. Without programs, a computer cannot do anything.

More Terms

- Language: A set of grammar which is used to define a solution to a problem.
  - Another word for **grammar** is **syntax**.
  - Making a grammar mistake in code is called a **syntax error**.
- Compiler: A tool which converts a solution into software, which a computer can understand.
- Operating System: Software which allows you, the user, to execute other software and interacts with hardware.

Hardware

Main Hardware Component Categories

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

Input Devices

Input devices provide information to a computer. What are some input devices?

Input Devices

Input devices provide information to a computer. What are some input devices?

- Keyboard
- Mouse
- Microphone
- Web camera
- Camera
- Joystick/Gamepad
- Drawing tablet
- Scanner

Output Devices

Output devices are used to represent the results of a process. What are some output devices?

Output Devices

Output devices are used to represent the results of a process. What are some output devices?

- Monitor
- Printer
- Speaker
- 3D Printer
- VR Goggles

Main Hardware Component Categories

1. Input devices feed into the CPU and Main Memory.
2. CPU and Main Memory will often work with Storage memory.
3. Results from CPU will feed into the 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

Main Memory

- Main Memory is **volatile**. In this context, **volatile** means that once the memory chip no longer has power, the data resets.
- Main Memory is called **RAM (Random Access Memory)**.
- Main Memory is a collection of logical gates.
- Each logic gate is called a **bit**. Each bit can retain an off or on position.
  - **off is usually interpreted as 0 or false**
  - **on is usually interpreted as 1 or true**
  - This is how we get binary code.
- A collection of 8 bits is called a **byte**.

Secondary Storage

- Secondary Storage is **non-volatile**: data is retained while the device is powered off.
- Examples of **non-volatile** memory:
  - Hard Drives
  - Optical: CD-ROM, DVD, Blu-ray
  - Flash Drives, Solid State Drives

Categories of Software

- **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

Programming Languages

Programs

- A program is a set of instructions that the computer follows to perform a task.
- We start with an algorithm, which is a set of well-defined steps.

Guiding the Green Circle

Pair up: Imagine that you have two objects on your screen. Guide the Green Circle to the Blue Circle.

- Blue Circle
  - Is Blue
  - Cannot Move
- Green Circle
  - Is Green
  - Has a direction
  - Can Move Forward 1 Unit
  - Can Turn Left 90 degrees
  - Can ask for Distance to Blue Circle

Machine Language

- Once we have an algorithm, we need to translate that algorithm into a programming language.
- After that, we ask the computer to translate that into a machine language.
- Computers only understand machine language.

Machine Language

- Machine instructions are short sequences of binary code that perform a low-level operation in the computer.
- Rather than writing programs in machine language, most programmers use programming languages.

Well-known programming languages

- C++
- C
- Java (You are here.)
- JavaScript
- Python
- Ruby
- Basic and Visual Basic
- Fortran
- C#

Where is Java used?

- Android Phone Applications
- Web Servers
- Desktop Applications
- Rarely is it used to make games, but there are some.
  - The best example of a Java game is Minecraft.

IDE: Integrated Development Environment

- An integrated development environment, or IDE, combine all the tools needed to write, compile, and debug a program into a single software application.
- IDEs make coding easier because they allow you to run your software with a single button click.
- Examples are NetBeans, Microsoft Visual Studio, Eclipse, CodeWarrior, etc.
- We will be using NetBeans in this camp.
  - NetBeans is written in Java.

What are Programming Languages Made Of?

What goes into a programming language?

- key words
- programmer-defined identifiers
- operators
- punctuation
- syntax

Programmer-defined identifiers

- Names made up by the programmer used to identify parts of the program.
- Not part of the language.
- Used to represent variables, methods, classes, and more.

Operators

- Used to perform operations on data.
- These are symbols and not words.
  - Arithmetic: $+$ - * /
  - Assignment: $=$

Other punctuation

- Other symbols used to organize the program.

Syntax

- The rules of grammar that must be followed when writing a program.
- This is the combination of everything we've seen in the previous few slides.

Variables

Variables

- In math, a variable is an unknown quantity. Sometimes you must solve for the variable in an expression.
- In computer programming, a variable is usually something which is known while the program is running but unknown when the programmer is writing the code.
- Imagine that you've been tasked to write the code which posts a status update someone's social media profile. (Think of Facebook or Twitter.)
- You don't know what a user will say when they use your code.
- After the program is written and is running, the user can decide what they would like to post.
- The message of the status update should be a variable.
- The variable will allocate sufficient memory to hold the user's message.

Variable Definitions

- In Java, you must write a variable definition (also called a variable declaration).
- The Java compiler will provide an error if it encounters a variable in an expression which has not been defined.
- There are many different sizes and types of variables and we will cover these in more detail in this course.

Input and Output

Input and Output

All programs from Microsoft Word to Facebook to Fortnight follow this general approach to processing data:

1. Get some data from an input device (such as the keyboard) or from a file.
2. Process that data using a combination of algorithm and secondary storage files.
3. Send the processed result to an output device (such as the monitor) or write the results to a file.

The Programming Process

The Programming Process

1. Clearly define what the program is to do.
2. Think a little.
3. Code a little.
4. Test. Return to Step 2 or stop if satisfied.

Fill the rest of the time.

Type into notepad your instructions for making a peanut butter and jelly sandwich.
If there is enough time, we'd like for you to tell us your instructions.