CSCE 110
Programming I

David Kebo Houngninou

Course Introduction

Instructions, algorithms, interpreter
What is the outcome of this course?

- Develop an understanding of programming
- Learn the Python programming language
- Understand the value of programming
- Appreciate the value of experimentation
- Acquire problem solving skills

To program is to solve problems
What is programming Language?

A programming language is a formal language that specifies a set of instructions that can be used to produce various kinds of output.

Programming languages generally consist of instructions for a computer. Programming languages can be used to create programs that implement specific algorithms.

Programmer languages classification

- **Computer Languages**
  - **Low Level Language** (Machine Language)
    - Use 1’s & 0’s to create instructions
    - Ex: Binary Language
  - **Middle Level Language** (Assembly Language)
    - Use mnemonics to create instructions
    - Assembly Language
  - **High Level Language**
    - Similar to human languages
    - COBOL, FORTRAN, BASIC, C, C++, JAVA
Programming languages classification

High-Level Languages (Basic, Java, FORTRAN)

low-Level Languages

Assembler

Compiler or Interpreter

Machine Code (Binary)

Hardware

What is the best way to learn a language?
What is an algorithm?

An algorithm is a finite sequence of steps that solves a problem.

Computational complexity:
How much computing resources are needed to solve a problem?
How long (time) and how much memory (space) does it take?
We observe the behavior of algorithms as the input size grows.

Problem

Given a list of positive numbers, return the largest number on the list.
How is code executed?

Code .c, .c++ ...
Compiler
Machine code 01010
Processor ARM
Processor Intel 8086
Processor IBM PowerPC

How is Java code executed?

Code .java
Compiler
Java ByteCode
JVM
Machine code 01010
Processor ARM
Processor Intel 8086
Processor IBM PowerPC
How is Python code executed?

Compiler vs. Interpreter

A compiler is a computer program that converts an entire program into binary code (machine code) targeted to a specific CPU.

An interpreter is a computer program that directly executes instructions written in a programming or scripting language, without compiling it into machine code.
**Interpreter vs. compiler**

Interpreting code is *different* from compiling code. The interpreter executes one instruction at a time.

<table>
<thead>
<tr>
<th>Interpreter</th>
<th>Compiler</th>
</tr>
</thead>
<tbody>
<tr>
<td>Translates program one statement at a time.</td>
<td>Scans the entire program and translates it into machine code.</td>
</tr>
<tr>
<td>No intermediate object code is generated</td>
<td>Generates intermediate object code</td>
</tr>
<tr>
<td>Continues translating the program until the first error occurs</td>
<td>It generates the error message only after scanning the whole program.</td>
</tr>
<tr>
<td>Python, Ruby, Perl, Matlab etc.</td>
<td>C, C++ etc.</td>
</tr>
</tbody>
</table>

**Why Python?**

- Easy to learn
- Solve problems in less time, fewer lines of code
- Versatile
- Elegant, intuitive syntax and dynamic typing
- Efficient high-level data structures
- Ideal for scripting and rapid application development
Applications of Python

Data Analysis
Machine learning
AI
Data Science
Web Applications

How to configure your computer?
1. Install the Python distribution

Anaconda is a free open source python distribution that provides packages and libraries out of the box for data science.

- The core python language
- Python packages/libraries
- Package manager

Install Anaconda (Python 3.7 version)
https://www.anaconda.com/download/

2. Install the IDE

An Integrated Development Environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development.

- An editor designed to handle code
- Build, execution, and debugging tools
- Version control

Install WingWare IDE 101
2. Install the IDE

WingWare 101 is Free.
It is available for major operating systems: Windows, MAC, Linux

About Python

Python was developed in 1989 by Guido van Rossum in the Netherlands.
Python was released for public distribution in early 1991.
How did Python begin?

van Rossum was having a hard time getting the job done with the existing tools available.

He envisioned that there was an easier way to get things done.

Python has been around for over 20 years, it is still relatively new to general software development.

Python has lots of support from the community.