An Overview of Programming Languages

Ruoyu Wang

Maths and Stats Programming Club
University College London

r.wang.13@ucl.ac.uk

October 15, 2014

Overview

Familiar Names

- Python
- C++
- Matlab
- HTML
- R
- VBA
- ...

Familiar Names

- Python
- C++
- Matlab
- HTML
- R
- VBA
- ...

An easy core question, what are the differences between Python and HTML?

Programming for different goals

General-Purpose Language

A general-purpose programming language is a programming language designed to be used for writing software in a wide variety of application domains. In many ways a general-purpose language only has this status because it does not include language constructs designed to be used within a specific application domain.

Programming for different goals

General-Purpose Language

A general-purpose programming language is a programming language designed to be used for writing software in a wide variety of application domains. In many ways a general-purpose language only has this status because it does not include language constructs designed to be used within a specific application domain.

Domain-Specific Language

A domain-specific language (DSL) is a computer language specialized to a particular application domain. e.g. web pages, matrix programming.

Programming for different goals

General-Purpose Language

A general-purpose programming language is a programming language designed to be used for writing software in a wide variety of application domains. In many ways a general-purpose language only has this status because it does not include language constructs designed to be used within a specific application domain.

Domain-Specific Language

A domain-specific language (DSL) is a computer language specialized to a particular application domain. e.g. web pages, matrix programming.

Non-strict definitions: Some languages may have specialized features for a particular domain but be applicable more broadly; or conversely may in principle be capable of broad application but in practice used primarily for a specific domain. e.g. R

Source Code

Source Code

In computing, source code is a script of computer instructions, written in human-readable computer languages. It cannot be excuted directly by the computer since the computer doesn't speak English!

Notably, we are programming in source code.

Source Code

Source Code

In computing, source code is a script of computer instructions, written in human-readable computer languages. It cannot be excuted directly by the computer since the computer doesn't speak English!

Notably, we are programming in source code.

Example (Python: Human-readable scripts)

```
print("Hi, you can understand what this code does.")
print("Because you know what 'print' means!")
```

Machine Code

Machine Code

Machine code or machine language is a set of instructions executed directly by a computer's central processing unit (CPU).
Machine code is unreadable. "Looking at a program written in machine language is vaguely comparable to looking at a DNA molecule atom by atom."

Machine code is for the computer to know what to do, not for us to read!

Machine Code

Machine Code

Machine code or machine language is a set of instructions executed directly by a computer's central processing unit (CPU).
Machine code is unreadable. "Looking at a program written in machine language is vaguely comparable to looking at a DNA molecule atom by atom."

Machine code is for the computer to know what to do, not for us to read!

Example (???)

```
000000 00001 00010 00110 00000 100000
100011 00011 01000 00000 00001 000100
000010 00000 00000 00000 10000 000000
```

Compliers & Interpreters

Compliers

Low-level languages like C++ often need to be compiled. A compiler is a piece of software that converts the instructions in the "source code" into (typically binary) machine code that can run by the hardware. A compiler thus turns your source code into an executable file (*.exe file) that can be run.

Interpreters

Python and higher-level languages are typically interpreted languages. This means that they do not need to be compiled into an executable. Instead a piece of software called an interpreter executes the instructions in the source code line-by-line directly into machine code.

Compiling into an *.exe typically leads to greater efficiency over interpreted languages although the efficiency gap has reduced over time.

Short list of General-Purpose Languages

- C
- C++
- Python
- Java
- JavaScript
- PHP
- Go
- Objective-C

Other noticable GPLs: BASIC, Pascal, Fortran, Haskell, Lisp, Lua, Perl, Ruby and Scala.

C & C++

C

- One of the most widely used programming languages of all time.
- Avaliablity on different operating systems.
- Ancestor of most modern popular GPLs.

C & C++

C

- One of the most widely used programming languages of all time.
- Avaliablity on different operating systems.
- Ancestor of most modern popular GPLs.

C++

- A superset of C, as 'plus plus' in its name.
- Popular & easy to access.
- Hard core, but high speed.

C & C++

C

- One of the most widely used programming languages of all time.
- Avaliablity on different operating systems.
- Ancestor of most modern popular GPLs.

C++

- A superset of C, as 'plus plus' in its name.
- Popular & easy to access.
- Hard core, but high speed.

Notably, if you want to learn C++, then you don't have to learn C first – Although C++ has inherited C's syntax, you can still learn C++ without C. They are two languages.

Python

- Young language.
- Comprehensible syntax, friendly to beginners.
- Many libraries.
- Multi-purpose.

Python

- Young language.
- Comprehensible syntax, friendly to beginners.
- Many libraries.
- Multi-purpose.

Remark: Python mainly has two versions at this moment: 2.7 & 3. Python 3 is an updated version of Python 2.7, but ironically it doesn't own compability with previous versions (You cannot run a Python 2.7 script on Python 3, vice versa.) . This property used to prevent a lot of users from adapting 3, for they wouldn't use a lot of useful libraries if they do so. The situation has been significantly improved. Nowadays most libraries are updated to Python 3.

Python VS. C++

What are the differences between Python and C++?

Python VS. C++

What are the differences between Python and C++?

- Python is much more beginner friendly and better in readability of codes.

Python VS. C++

What are the differences between Python and C++?

- Python is much more beginner friendly and better in readability of codes.
- C++ is strict in syntax: You have to add a ; at the end of every sentences, and can't forget about quotes {}.

Python VS. C++

What are the differences between Python and C++?

- Python is much more beginner friendly and better in readability of codes.
- C++ is strict in syntax: You have to add a ; at the end of every sentences, and can't forget about quotes {}.
- But Python runs slower than C++, for the reduction in efficiency brought by the interpreter.

Python VS. C++

Example (C++: Hello World)

```
#include <iostream>

int main()
{
  std::cout << "Hello World!";
}
```

Python VS. C++

Example (C++: Hello World)

```
#include <iostream>

int main()
{
  std::cout << "Hello World!";
}
```

Example (Python: Hello World)

```
print("Hello, World!")
```

Short list of Domain-Specific Languages

- Web design: HTML, CSS
- Matrix programming: Matlab
- Symbolic mathematics: Mathematica, Maple
- Statistical modelling: R, Stata
- Markup language: LaTeX
- Database queries: SQL
- Spread sheets: VBA

Python VS. Matlab or Mathematica

Python is different from Matlab or Mathematica in serveral ways:

Python VS. Matlab or Mathematica

Python is different from Matlab or Mathematica in serveral ways:

- Python is a GPL, indicating that it can easily handle some 'non-computing' things, such as reading from or writing into a file, or the manipulation of strings.

Python VS. Matlab or Mathematica

Python is different from Matlab or Mathematica in serveral ways:

- Python is a GPL, indicating that it can easily handle some 'non-computing' things, such as reading from or writing into a file, or the manipulation of strings.
- Matlab and Mathematica are more specialising in numerical computing and symbolic mathematics, respectively. They have got really fancy toolboxes for some special use.

Python VS. Matlab or Mathematica

Python is different from Matlab or Mathematica in serveral ways:

- Python is a GPL, indicating that it can easily handle some 'non-computing' things, such as reading from or writing into a file, or the manipulation of strings.
- Matlab and Mathematica are more specialising in numerical computing and symbolic mathematics, respectively. They have got really fancy toolboxes for some special use.
- But you can refer to two libraries in Python to do the same thing as they do, namely Numpy and Sympy!

Python VS. Matlab or Mathematica

Python is different from Matlab or Mathematica in serveral ways:

- Python is a GPL, indicating that it can easily handle some 'non-computing' things, such as reading from or writing into a file, or the manipulation of strings.
- Matlab and Mathematica are more specialising in numerical computing and symbolic mathematics, respectively. They have got really fancy toolboxes for some special use.
- But you can refer to two libraries in Python to do the same thing as they do, namely Numpy and Sympy!
- Numpy and Sympy are all open source and free.

Application: What I make use of DSLs

I myself am using these languages in useful ways, for your information:

Application: What I make use of DSLs

I myself am using these languages in useful ways, for your information:

- I used HTML, CSS and JavaScript to write my own website. `http://rwang.org/`

Application: What I make use of DSLs

I myself am using these languages in useful ways, for your information:

- I used HTML, CSS and JavaScript to write my own website. `http://rwang.org/`
- LaTeXfor my academical essays, this slide (package beamer), and even my CV!

Application: What I make use of DSLs

I myself am using these languages in useful ways, for your information:

- I used HTML, CSS and JavaScript to write my own website. `http://rwang.org/`
- LaTeXfor my academical essays, this slide (package beamer), and even my CV!
- I am doing maths so sometimes I use Mathematica for some symbolic calculations.

The End