Andrew Sinn

Programming Language Concepts

Professor Brown

11/06/2011

Project #4 - Python

First of all, I would like to say that I took the Computer Science introductory sequence when the language taught was Java and not in Python. With that being said, I never learned Python and decided to choose this language so that I could possibly help the freshman through the introductory sequence of Python.

Python first debuted in 1991 and it was developed by Guido van Rossum[[1]](#footnote-1). Python had underwent through three major updates in it’s lifetime. With its updates, it has added functional programming tools such as lambda calculus, garbage collection, and backwards compatibility with older versions. Right now, the current version of Python is 3.2.2.

Python is an open-source, high-level interpreted language that has many advantages when it is chosen as the programming language of choice to solve a problem. One of the advantages of Python is that the syntax and code is very readable[[2]](#footnote-2). Since the language is high level, the syntax looks almost like the English language, which makes it easy for coders and readers to understand what is being done in a program. The language also makes indentation standard in the language when it comes to control statements which can help in identifying logical flow in code throughout the program. The language does away with braces or any other delimiter that signifies a block of code in a control statement. Python instead uses forced indentation or tabbing to separate code in control statements. This forced indentation fixes the dangling else problem by requiring nested if statements to be indented[[3]](#footnote-3). With the high-level syntax and forced indentation, it allows for people that are new to Python to pick up the language easily as opposed to other languages. This is a reason why some schools have considered using Python as an introductory language for Computer Science programs (like RIT).

Another advantage of using Python is that the standard library is huge. The library has many built in functions that can help solve problems in almost any domain. With the huge library, Python can also solve problems in different programming paradigms. As mentioned before, Python has functions that support the functional programming paradigm. It also supports the imperative and object-oriented paradigms. Being flexible with paradigms and having a huge library makes a case for Python to be chosen as the programming language to code in.

Another advantage of Python is the portability of the code. Python can be run on Windows, Linux, Mac, and other operating systems. It can also run the .NET framework and the Java Virtual Machine via specific ports of the language to those platforms. Thus, it makes the language flexible on most machines currently.

Python handles parameter passing via pass by reference. Thus, arguments that are passed in are references to that argument and will be changed if modified in the function that had the argument passed in. Types in Python are defined by values. Values can possibly be strings, numbers, tuples, lists, and sets. The values determine what types they are. Thus, Python supports dynamic typing as types are not checked until runtime. Python is also strongly typed: it does not allow type mismatching at runtime. It will result in an exception being thrown at runtime. Strings are also immutable in this language.

Scoping in Python is static[[4]](#footnote-4). Variables will be associated with the innermost scope first. Then it will be associated outward to outside function scopes. Then scope will be associated outward towards the global scope. The last scope is the namespace scope in which variables can be associated with.

With all the advantages in Python: readability, easy to learn, huge library, multi-paradigm, and portability, it can be seen why Python could be a good choice to solve a programming problem. With these advantages, Python can continue to grow and can offer more options to new and veteran programmers.

1. [Python's Log History](http://svn.python.org/view/*checkout*/python/trunk/Misc/HISTORY) [↑](#footnote-ref-1)
2. [About Python](http://www.python.org/about/) [↑](#footnote-ref-2)
3. [Dangling Else](http://docs.python.org/reference/compound_stmts.html) [↑](#footnote-ref-3)
4. [Python Scoping](http://docs.python.org/py3k/tutorial/classes.html%23python-scopes-and-namespaces) [↑](#footnote-ref-4)