PYTHON: A CRITICAL ANALYSIS OF PROGRAMING LANGUAGES FOR NOVICES

[1] Rijja Hussain Bukhari, [2]Aaliya Sarfaraz, [3]Ahmed Khan
[1]Department of Bio Sciences, [2, 3] Department of Computer Sciences
[1,2,3] COMSATS Institute of Information Technology (Main), Islamabad, Pakistan.
[1]rijjahussain@gmail.com, {[2]aaliya.sarfaraz, [3]ahmed.khan }@comsats.edu.pk
Corresponding Author Email: ahmed.khan@comsats.edu.pk

***ABSTRACT*:** *The programming languages are developing science they were developed. The rapid changing trends in the use of new programming languages in industry is not maintained in academia where a beginner listen in programming is always from C and Java although students must learn advanced languages before entering industry, which bring us to concerned that it is one the major reasons that student are intimidated by coding. There are several researchers that have compare the programming language using different standards and requirement for better programming language but most of them have agreed that C and Java have complex syntax whereas Python has simpler syntax that make it easy and fast to learn.*

Keywords Python, programming languages, comparison for programming languages, Java and python, python standard libraries.

INTRODUCTION

The programming languages have many definitions one of them describe the programming language as a collection of variables and data structure which have a follow of execution [11]. C is one the early programming language that was developed by scientist C is a procedural programming language that require low memory and less runtime support. It uses straight forward compliers [12]. After sometime there were advancement made in C which resulted in C++ C++ can be thought of as object oriented version of C. But as the requirement grew the use of C and C++ was not sufficient for effective computation so Java language was initiated java language java is a general purpose language that is also object-oriented and it is purely class based. That java always work in classes and objects. Since than many languages have been developed one of them is Python. Python is also general purpose language that is easy to understand because of its simple coding syntax [13]. Python has many versions that uses the binding of python with other languages for example java and python Jpython and C and python Cpython [14]. But the concerned is that if computer languages are developing so rapidly and with this rapid development programming languages that are taught at beginners courses should also be changed according to changing trend in the use of language in the industry. But that is problem still many beginners start coding in C or C++ or java which is complex and lead many students to the idea that coding languages is the mostly about learning syntax that not only result in decline in developed of better codes but also is the reason mostly student can code a small code but cannot transform that into a working application. To analyze how one programming language is easier to learn from other some scientist work was reviewed that showed Python is easier to understand and faster to learn.

LITERATURE REVIEW

In [1] Perchelt, represented the idea that often debates about the better programming language tends to be inconclusive in the end so, Perchelt have try to solve this debate by empirical comparison of seven different languages i.e. C, C++, Java, Perl, Python, Tcl and Rexx. Perchelt starts his study by implementation of the same program in all the languages and then Perchelt analyzed different implementation by the different programmers. This removes the biased due the implementation by the single programmer and allow the comparison of variability of different program properties each langue produce. According to Perchelt there two main categories of language among these seven languages. Scripting languages like Perl, Python, Rexx and Tcl that focus on interpretation rather than compilation whereas Conventional languages like C, C++, Java focus on less interpretation and more on compilation. Perchelt, represented the results data in the boxplot, to give the clear visual representation of the comparison of languages. Perchelt also used bad-to-good ratio, Median comparison and Wilcoxon rank sum test and results of each of these test is a p value. Perchelt run program to test seven main properties i.e. runtime, initialization phase, memory consumption. program structure, work time and productivity, program reliability and length and amount of commenting. Perchelt study concludes that constructing and writing script languages, Python and Perl run faster than Tcl for both phases that are initialization of phenocode and searching through data structure while C and C++ was faster than Java and scripts. Perchelt study is very generalized and had evident conclusion but their limitation due to restricted number of implementations and programmers investigated for implementation. The resulting groups off language may not hold when applied to different problem. In [2] Millman, et.al present the idea that for past decades python has emerged and used in scientific research. Millman et.al describes the history of python that although python come with its various type of build-in data types but still it was not sufficient to use for numerical computing, so matrix-sig was developed. It was a group of special interest that worked on developing array data type in python. Jim hugunin developed C-extension module called Numeric. It was based of Jim felton's extension and ideas from matrix-sig, but soon after he left to develop python implementation

Table:1 table for comparison programming language [16].

Language	Safety	Expression	Type compatibility and equivalence	Checking
C	weak	explicit	nominal	Static
C++	weak	explicit	nominal	Static
Java	strong	explicit	nominal	Static
Python	strong	implicit (with optional explicit typing as of 3.5)	structural	dynamic

Table 2. Programming-language comparison statistics[1]

Language	Number of programs	Compiler or execution platform
Tcl	10	Tcl 8.2.2
Rexx	4	Regina 0.08g
Python	13	Python 1.5.2
Perl	13	Perl 5.005_02
Java	24	Sun JDK 1.2.1/1.2.2
C++	11	GNU g++ 2.7.2
C	5	GNU gcc 2.7.

using java called JPython as mentioned in Table 1 and Table 2. The interaction between the python and scientific communities resulted into python gaining syntax it required for research by scientific community. These changes include syntax changes in slicing and accessing array elements through index. By the next few years Travis Oliphant, Eric Jones, and Pearu Peterson, merge their codes and called the resulting module SciPy which increase the usefulness of Python. Fernando Pére released the latest version of IPython which is the interactive shell used by scientist. Millman et.al states that the Travis Oliphant released NumPy in 2006 which help to grow the SciPy since. Further, Millman et.al identifies the use of python for mathematical programming. Three major projects are discussed by Millman et.al, in [2]. Simply is one of the algebra system written purely in python. mpMath is a library of python that is used for multi precision float-point arithmetic Sage is software that open source and provide all the mathematical methods of python at one place.in the end Millman et.al, concludes by saying that a good programming language is the one that not provide great mathematical computation but also easy to use and is flexible enough to be used in web based software and provide facilitation of database related application.it should be package having graphical and syntaxial ease. In [3] Oliphant, presents that it's not new that a language that is high level increase the productivity and incase of python it's one of the language that is close to pseudocode which make it easier to implements large algorithm that involve large data sets. Oliphant begin by mentioning major advantages of python liberal open license, clean syntax, interactive interpreter, easy to embed python in any application, ability to interact with any software, large number of library module, having community that quickly response to user queries and easy to available and install software. Oliphant discuss the clear syntax of python which make its use as programming language for scientific computing easy and maintainable. Python feature of dynamic programming make it easier to

define and use function. Oliphant says, everything in Python is an object of a kind. The built-in data types are integer, floating point, and complex. The interactor of python make implementation even more simpler. Python also have a clear separation of logical groups of codes like Modules, classes, functions. Python also make it easy to code as may standard library that come with python include csv, re, zlib/tarlib, gzip, decimal, pickle, ctype, os, sys. Python provide the flexibility of call pythons methods from C which provide the python extension is C. This is amazing feature that combine two languages without changing the syntax of any one of them. The input/output (io) sub package, the linear algebra (linalg), statistics (stats), optimization (optimize), integration (integrate), interpolation (interpolate) The Fourier transforms (fftpack), special functions (special), sparse (sparse, linsolve), Nd-image (ndimage), signals and systems (signal) sub package which provide multiple functionalities.Weave is also one of the module of Python that can be used to call C code from interface of python as well as useful for bulding modules of extention. Thus, Oliphant concludes that Python build in libraries as well as SciPy and NumPy provide ease of functions and coding in python.

In [4] Milano et.al, starts by saying that a good programming language should stratify the needs of scientific studies. Python is safe to type and dynamically used programming language that have feature of polymorphism, multiple processing and simple implementation. Python also support multiple threading and parallel processing. Python is a latest language that is based on classes structure. Its Numpy library is used for multipurpose use of arrays. Numpy make multidimensional array easy for manipulation. Data file parsing and reducing the power flow problems and soring the output in the right file are major attributes of Python. These attributes are gifts of many modules provided by python. Each module must be fulfill the communication standards for

the core functions. Milano et.al, illustrated a case study to help us understand these features which can described as the capability of including all the libraries to provide efficiency in matrices and due to this DOME can be used to evaluate mathematical modules and their suitability for analysis of power system. Ability of mathematically integrating statistical differential equations for this reason many devices using statistical model were included. Multiple processing was also used to run these stimulations. Ability to solve eight digit study delayed differential equations. This evaluation is very significant as some calculations can be extensive for small systems and because study was comparing many python modules. In [4] Milano et.al, concludes by saying that latest, developed, complete and diverse scripting language that is complete appropriate for scientific research and study of power system analysis. The case study presented in paper presents the complete connection of python with the general purpose mathematical modules that provide efficient calculation of power system analysis. In [5] Perez et.al, presents that over last ten years, high level programming, easily implemented languages with multiple modules and great graphical feature are famously used in the field of computer sciences. Python interacting platform provide the help with of its documentation but its very simple and easy to use platform. Processing of regular expression, network processing on a low level, parsing of XML, web connecting services are all provided by different modules of python.by the end 2001 IPython was developed with purpose to improve the interacting platform that is used in the default language and extend the abilities of this platform to get an advanced working environment. IPython is an open source tool that have license and runs in the environment of linux,unix,Mac,windows operating system. IPython is best suitable interacting platform for everyday task. The common python platform stores the last variable assignment of user in memory while IPython provide user with options to choose from which point the result need to be stored. IPython have set of commands that be used by user to improve the interaction between the system and user. These commands are inspired by tradition unix interacting shell where separation and options are indicated. This is accessible to the end user who can change the command. The IPython provides user with documentation of python and complete ability to acees the object name by using Tab key from keyboard. IPython provide a interaction platform that trace the complete path of the running code so that user can check the step by step faults in the crashed code. The IPython new designed consist of IPython core which shows the IPython core function as python library as compare to the terminal based python application. In [5] Perez et.al, presents that IPython provides a better interacting extension of original python shell with the interacting attributes and tools so that scientific programming become easier. In [6] Van Der Walt et.al, present the idea that NumPy is one of the most commonly used module of python. the biggest reason of vast use is the amazing facilitation of array and mathematical function. Van Der Walt et.al, describe the NumPy array as a collection of elements which is differentiated by the type of element and shape it have. NumPy can support up to 32 dimension array with other kinds of elements or even combinations of elements. To use NumPy just import the module in the iterative mode of python and after that you can easily use it anywhere in code. NumPy have different fascinating features which include data pointers (memory address of and array), datatype description (what type of element stored in array), shape (grids stored in array of x, y and z axis) strides(the bytes processor skip before dealing with next element member of array) and flag that show if we are allowed to modify the array or not. Vectorization is feature which combine these feature perform on each element in group so that better manipulation of arrays can be done. If NumPy vectorization is implemented in C noticeable difference in speed was observed. Firstly it is check that whether two arrays matched or not and two arrays only match if they are equal in length. In other case array with smaller sequence is expand to the length of larger one. Memory mapping is technique that is useful if we want to map arrays with small length because arrays is stored on disk and can be directly addressed with copying it into memory. Arrays of NumPy are called homogeneous if they contain element that have same datatype. Structured array are the NumPy arrays with compound elements. In [6] Van Der Walt et.al, concludes by saying that NumPy provides a platform that make numerical problem solving using high level code easy which make is simple to use for scientific computing.

In [7] Lo et.al, focus on the subject that which programming language is better to teach a computer science student first. It is very interesting insight on the which programming language is easy to learn and understand from a student point view. In [7] Lo et.al, says that mostly beginners of programming course start learning the language syntax and mostly try to write code keeping mind the syntax details which results into a problem that is they can write a small cod for dose not know how to convert it into a complete program. Mostly the first Programming language taught and learn is java but often time students find difficulty in understanding complex concepts like classes, abstract, methods etc. According to Lo et al, the first programming language taught should have features such as :(1) Simple input/output statements, (2) Readable and consistent syntax, (3) Clear syntax, and (4) Orthogonality. To investigate which programming language is better and easy to learn by students Lo et al, conducted an experimental research in which included the designing of related questionnaire, asking the questions from students in university who have learn java or python or both for at least half year and teachers who are also familiar with the language. The questionnaire evaluates Programming concept comprehension which is students' comprehension and perceptions about data types, expressions and assignment, the control structure, the loop structure, function, and objects is examined. Program implementation which is to understand whether students can successfully write programs using suitable data types, structures, functions, or objects and Cognitive load which intend to examine students' cognitive load in learning the programming concepts and skills. Although [7] Lo et.al, does not share the result of this experiment but still the research presented a different view for discovering the better programming language i.e. to consult the learner instead of an expert.

In [8] Cottom, provides the idea that due to increasing approach of scientific computing to combine python with compiled module to improve performance. In [8] Cottom focus on binding SWIG with Python. SWIG provide the C

and C++ libraries in other languages such as python after process the code in C language. In [8] Cottom says that it is better to use header file if quick results are needed but information specification explicitly provide more control over the interface. Most of the time interface firstly define python's module name, then it will establish the rules that are applied to module and then at the end python specifies the binding definition. One can either install the shared module as well as the Python source module in the place available to the Python program, or mention the module location in the interactive platform. Although by binding SWIG and python there may be some function which share the same name but different set of arguments which may lead to function overloading. SWIG solve this by binding python public classes and function where private and protected classes remain same because they are not binds. As C++ is nor restricting than Python so python also provide key board arguments facility. So, SWIG provides instant key board argument ability to python bindings. C++ and Python have related operators. The thrown exception C++ are displayed as a runtime error in python, strings for complex type, numeric for numeric type and exact message in string type exceptions. In [8] Cottom concludes that SWIG efforts to provide function overloading, key arguments facilitation as well as translation of exceptions.

In [9] Lindstrom, presents the idea that IT professional that work in any field have the same requirements for a programming language. He says that he used python because its powerful, well supported and easy to use. According to Lindstrom we need new and advanced programming languages because computer science is constantly growing and with the advanced hardware and faster techniques the new programming languages are quickly replacing the old ones. In [9] Lindstrom view python feature that is simple syntax, function based on network and building library for graphical interaction and especially ease to learn make it a better choice Lindstrom continue by sharing with us the examples of people who are using python like Tommy Burnette, senior technical director at Industrial Light & Magic, Peter Norvich, director of search quality and many more. Team developing python has slow-growth policy and extensive study before addition in language. He emphasized that python has developed in communities that are helpful and cooperative. Python other efforts include that python takes uses of the C libraries everywhere in systems, identifying the team of developers associated with java libraries, there are some python developers that combine the java libraries with python to create JPython. Complex programming is a distinct feature of Python because Pythons syntax is simple and built-in data structures allow user to focus layout's complex features only. Python is easy to learn because one can find several tutorials from python's homepage there are also many books that provide simple steps to learn python moreover there are websites such as Useless Python that have different task that can be used to practice python. In [10] Perez et al, refer to the python as a scientific ecosystem the Python have several open source tools for computation that are developed by many scientists to make it easier for scientific work. Python tools try to provide well structured data types rather than changing the algorithms and code in the different versions. There are other languages like Fortran and Matlab that were designed mainly for mathematical computations, so their interactive platform is hard to understand, and they do not provide flexibility for every day computations. Guido van Rossum who developed Python have given details about the language history in any articles. He had a desire of python being the language that is easy to interpret and write but also high level. He wanted Python to be different from java in all manners. Pythons offers the low-level capability, portability and define error model based on exceptions. Python provides its user easy handling of complex programming function which also concise. Python is very suitable for developing tools that specialize to only scientific research and work. NumPy is very efficient and appropriate for scientific calculations that is also available in C and Fortran that improve their speed and provide interaction like python interaction. In [10] Perez et.al, concludes that python has matured to a level where is highly suitable for scientific computing. The use of python is increasing due to the efforts of developers that making great improvements in python [17-24].

CRITICAL ANALYSIS

After extensive review of the above articles it's not wrong to say that all the programming language have pros and cons but python simplicity and closer to human understanding syntax has made it to shine from all other languages. As academia is the basis for scientific community evolution it's important that student at the beginning of their computer science career learn a language that does not involve complex so that student starts coding by focusing on semantics rather than syntax. Python has tremendous advantages that make it interesting to learn and taught. First and foremost, python homepage has several tutorials that can be easily accessed by the students and teachers, now they find themselves stuck during development of codes. Secondly python is the only language that is closer to human language one can easily interpret meaning function and classes. Further python comes with batteries included which means it has already install all the necessary modules required for a better programming.

Today programming has become the second language to excel in any field, so a programming taught to novices should be flexible enough to be integrated at any field not just for someone working in software houses. When we take programming language as subject it is important to understand that a subject should be taught in a way that match the student psyche. If the first language that a student learn is C or Java than it is often fund that student start focusing on remembering the syntax rather than learning how to solve problem. Python has modules that make it easy to handle, analyze and manipulate data on a large scale. The famous NumPy as in [5] called as the efficient numeral computation structure is providing students with amazing ways to ease the computation of mathematical problems that is not seen in any other language. Matplotlib is another module that transform that tabular data into graphical images which make statistical analysis of huge amount of data much easier. SciPy the library that has brought many scientists from ruby to python due to its vast algorithms and tools for mathematical computation. Twisted a library for network application

development and comes with very beautiful api. SQLAlchemy a library for database related work. Scrapy a library that is best if one is interested in scraping data from the web. Requests written by Kenneth reitz is the most important http library for python developers. In a nut shell one can say about the python libraries that anything you name python has a library for it. As in [10] Perez and [3] Oliphant discussed that python provides the best atmosphere for scientific programming. Python's features are the reasons that attract scientist from other programming languages to python. So why not start the learning python first. Both have view which shows that today scientific community believe that python provide better platform for easer programming and variety of functions that are required by scientist in all fields. If we combine these believes with Perchelt [1] thoughts than one can say that python is the need of hour and should be consider as first language to be learned by novices so that it is easier for them to translate their solution into a working application.

CONCLUSION

The programming languages has always been developing according to the hour of need and these has brought us very forward from where we started and as the needs will change the merits for better programming languages will also change so it is important to understand the importance of language taught to novices in computer sciences and according our study of some scientist views on programming languages comparison we conclude that python is best suited. As python has the simplest syntax it will make students familiar with programming faster and help them emphasis on problem solving and critical thinking rather than being stuck in complex syntax. Although this work was limited in research and resources there are people and papers that presents the thoughts which may imply different.

REFERENCES

[1]. L. Prechelt, "An Empirical Comparison of Seven Programming Languages", Computing Pratices IEEE, pp. 23-29, October 2000.

[2]. K. J. Millman and M. Aivazis, "Python for Scientists and Engineers," in Computing in Science & Engineering, vol. 13, no. 2, pp. 9-12, March-April 2011.

[3]. T. E. Oliphant, "Python for Scientific Computing," in Computing in Science & Engineering, vol. 9, no. 3, pp. 10-20, May-June 2007.

[4]. F. Milano: "A Python-based Software Tool for Power System Analysis", IEEE PES General Meeting 2013, Vancouver, Canada, July 2013.

[5]. F. Pérez and B. E. Granger, "IPython: a System for Interactive Scientific Computing", Comput. Sci. Eng., Vol. 9, No. 3, pp. 21-29, May 2007.

[6]. S. Van Der Walt, S. Colbert, G. Varoquaux, "The numpy array: A structure for efficient numerical computation", Computing in Science Engineering, vol. 13, no. 2, pp. 22-30, March 2011.

[7]. C. A. Lo, Y. T. Lin and C. C. Wu, "Which Programming Language Should Students Learn First? A Comparison of Java and Python," 2015 International Conference on Learning and Teaching in Computing and Engineering, Taipei, 2015, pp. 225-226.

[8]. T. L. Cottom, "Using SWIG to bind C++ to Python," in Computing in Science & Engineering, vol. 5, no. 2, pp. 88-97, Mar/Apr 2003.

[9]. G. Lindstrom, "Programming with Python," in IT Professional, vol. 7, no. 5, pp. 10-16, Sept.-Oct. 2005.

[10]. F. Perez, B. E. Granger and J. D. Hunter, "Python: An Ecosystem for Scientific Computing," in Computing in Science & Engineering, vol. 13, no. 2, pp. 13-21, March-April 2011.

[11]. Pierce, Benjamin (2002). Types and Programming Languages. MIT Press. p. 339

[12]. Lawlis, Patricia K. (August 1997). "Guidelines for Choosing a Computer Language: Support for the Visionary Organization". Ada Information Clearinghouse. Retrieved 18 July 2006.

[13]. Gosling, James; Joy, Bill; Steele, Guy; Bracha, Gilad; Buckley, Alex (2014). The Java® Language Specification

[14]. McConnell, Steve (30 November 2009). Code Complete, p. 100.

[15]. "History and License". Retrieved 5 December 2016. "All Python releases are Open Source"

[16]. Felleisen, Matthias. On the Expressive Power of Programming Languages. ESOP '90 3rd European Symposium on Programming.

[17]. Khan A, Sarfaraz A. (2017). "Vetting the Security of Mobile Applications", Science International, 29(2), 361-365.

[18]. Khan A, Sohaib M, Amjad F M. (2016). "High-Capacity Multi-layer Framework for Highly Robust Textual Steganography", Science International, 28(5), 4451-4457.

[19]. Khan A, Tariq U, Shabbir J, Hassan S. (2016). "Cloud Security Analysis for Health Care Systems", International Journal of Computer and Communication System Engineering, 3(1), 1-8.

[20]. Azeem S, Khan A, Qamar E, Tariq U, Shabbir J. (2016). "A Survey: Different Loss-less Compression Techniques", International Journal of Technology & Research.,4(1),1-4.

[21]. Shah M, Khan A. (2015). "Implementing User Authentication Service for Cloud Network", Science International, 28(6), 5301-5306.

[22]. Khan A. (2015). "Comparative Analysis of Watermarking Techniques", Science International, 27(6), 6091-6096.

[23]. Khan A. (2015). "Robust Textual Steganography", Journal of Science, 4(4), 426-434.

[24]. Khadim U, Khan A, Ahmad B, Khan A. (2015). "Information Hiding in Text to Improve Performance for Word Document", International Journal of Technology and Research, 3(3), 50-55.