JECET; December 2019- February 2020; Sec. B; Vol.9. No.1, 015-020. . E-ISSN: 2278–179X
[DOI: 10.24214/jecet.B.9.1. 01520.]

Journal of Environmental Science, Computer Science and Engineering & Technology

An International Peer Review E-3 Journal of Sciences and Technology

Available online at www.jecet.org

Section B: Computer Science

Research Article

Juxtaposing Python with BASIC in the Context of Introductory Programming

[1]**Douglas, T. Minafa-a and** [2]**Ojekudo, A. Nathaniel**

Department of Computer Science, Faculty of Natural and Applied Sciences, Post Graduate School, Ignatius Ajuru University of Education, Nigeria.

Received: 23 November 2019; **Revised:** 06 December 2019; **Accepted:** 11 December 2019

Abstract: Computer programming and programming language is relatively machine-dependent, as it spring up control element called software through computational algorithms. Introductory programming is quite fundamental to study of computing and computational theories, and as such it is offered /taught at elementary or intermediate level for computer science majors across tertiary institutions. Though, different programming languages had came into being with focus and diversity for problem solving in different domains. Exploratory context of lexical and semantic similarities between BASIC and Python were provided in this paper.

Keywords: Introductory Programming, Language, Computer Science, Python, BASIC

1. INTRODUCTION

Programming is concerned with the design principles and development process of developing system programs and software that facilitate operational control in computing gadgets [1]. Programming requires effective tools and technical understanding of program development, computer is designed to receive users' input, execute and produce output [2].].

Program is the sequence of standardized instructions and predefined process for task execution on request [3]. A process could comprise of multiple threads that execute instructions concurrently. Compiler is a program that handles the compilation routine of computer program, and the program flowchart presents the logical steps and operational procedure of program [4]. System programming is a pragmatic approach within the field of computing, which involves code creation [5]. A computer program is a logical collection of instructions and interdependent execution with predefined processes, multiple processes may be associated with the same program; for example opening up several instances of the same program often means more than one process is being executed commonly regarded as multitasking[6]. Compilation approach is associated with the structure and elements of computer program because the execution attention is more on the operation than the required operands for computational process [7,8].

Basically, computing gadgets are built with physical components; their structural and logical design is electronic and mechanical in nature [7].

However, selection of a programming language for teaching programming fundamentals to computer science majors at introductory level hinged on certain factors as shown in **figure 1**.

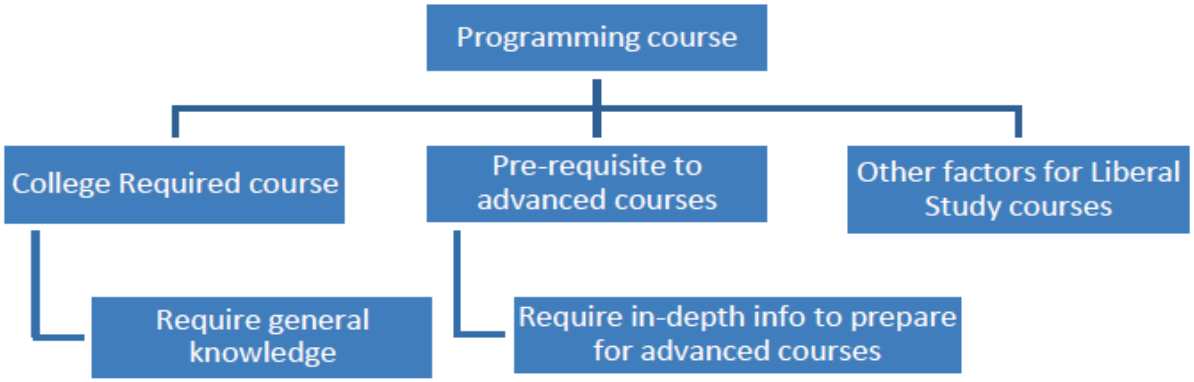

Figure 1: Determinants for Programming Course Design [1].

2. THEORETICAL REVIEW

Language is generally defined as human form of communication that uses peculiar traits, such as voice, gestures, or symbols. It is a method of human interaction using verbal or non verbal mode; the particular words and speech that is used by the people of a country, area or social group [10].

Human-Machine Communication: Language is also the modality or processes used to ensure an agreement between the sender and receiver for interpretations and the schema for combining their use for communication. Language lies majorly on human capacity for acquiring and using complex logic of

DOI: 10.24214/jecet.B.9.1.01520.

communication in certain circumstances [9]; this is referred to as natural language because it emanates from human entities, with transition to machine understanding called formal language.

Essence of Programming Language: A programming language is a notation for coding programs, which are specifications of a computation or algorithm. A programming language is code editor for creating other programs because it provides machine with syntactic and semantic production of the target program [6].

A programming language is a notation used to script programs, which allows computer to perform series of computation through algorithms [10]. Programming language could also be perceived as syntactic method of combining words in written form with which instructions are coded[11,12]. Every programming language has its own compiler for execution. Consequently, different programming languages had came into being with focus and diversity for problem solving in different domains or real world applications. Hence, a typical programming language may not be suitable for all purposes or problem domains; because every programming language has it syntactic structure and compiler for executing instruction set given by programmer[11] .

Syntactic layout of Python programming language posed so much similarity with BASIC, in terms of building block and code fragments. Python does not mandate instruction terminator, just as BASIC presumed every statement on separate lines to be executed in isolation without the need for end of statement tokens and separators. Despite the fact that BASIC is a common choice for teaching introductory computer programming by most instructors / computer educators, just because it had been in existence for long; both languages should provided comprehensive set of constructs and APIs that serve general purpose as well as specific programming tasks.

They both support primitive data structure and user defined data types and community support by documentations. BASIC was originally high level language before technological turnaround with Visual BASIC from Microsoft, whereas the lexical construction of Python fit-into high level and object oriented language, thereby having powerful syntax with utmost simplicity for beginners. Python shell makes it possible to execute instructions from console prompt or code editor, because it shares operating system and machine compatibility with BASIC to run on most personal computers and variety of platforms with minimum or very mild system configurations.

3. PYTHON AND BASIC OVERVIEW

Python programming language was developed in 1988 by Guido Van Rossum, while BASIC programming language was developed by Thomas Kurtz and John Kennedy in 1964. BASIC is actually an acronym for Beginners All-purpose Symbolic Instruction Code; a high level language (HLL) that is suitable for business and scientific task. Python on the other hand, emanated as fourth (4^{th}) generation language (4GL), and can as well serve all purposes. In contrast to other commonly used languages for software development, it strives to inculcate syntactic simplicity.

Python is currently being used for application program development at international information technology organizations like Google and Yahoo. Practicing programmers and intermediate developers

DOI: 10.24214/jecet.B.9.1.01520.

can accomplish complex task with Python, meanwhile Python capability and robust features are also accessible to programming beginners to handle more interesting problems.

The rigidity perception in coding and syntactic structure for most programming languages had been the points of difficulty in teaching and learning computer programming at introductory level because it does not allow maneuver and deviation. Teaching of programming skills and computer language within the university context seem to be very complex and challenging task. Programming is a fundamental course taught to computer science learners during undergraduate studies. Hence, the selection of a programming language for training computer science students is of great importance. Computer science educators have expressed concern over the difficulties with which beginning programmers demonstrate the logic and principles of programming.

4. ILLUSTRATIVE COMPARISON

Quick BASIC (QB) is often chosen as language translator for executing BASIC program; just as IDLE comes as simple IDE for executing Python program. Language translator for BASIC and Python inculcates in-built interpreter that check program statements line by line for lexical and semantic compliance. Simplicity, reliability, data structure, machine compliance and other factors for adopting BASIC in teaching introductory programming course, are applicable to Python as well; just as their lexical pattern and coding seem similar as depicted in these figures:

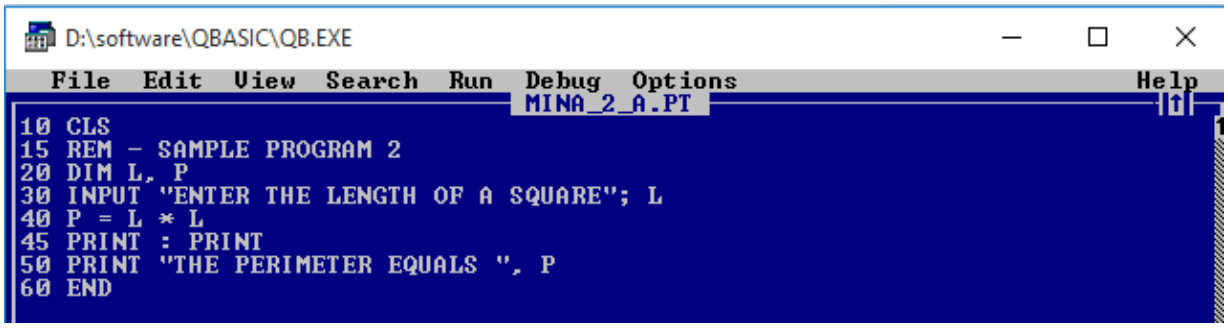

Figure 2: BASIC source program – Code Fragment

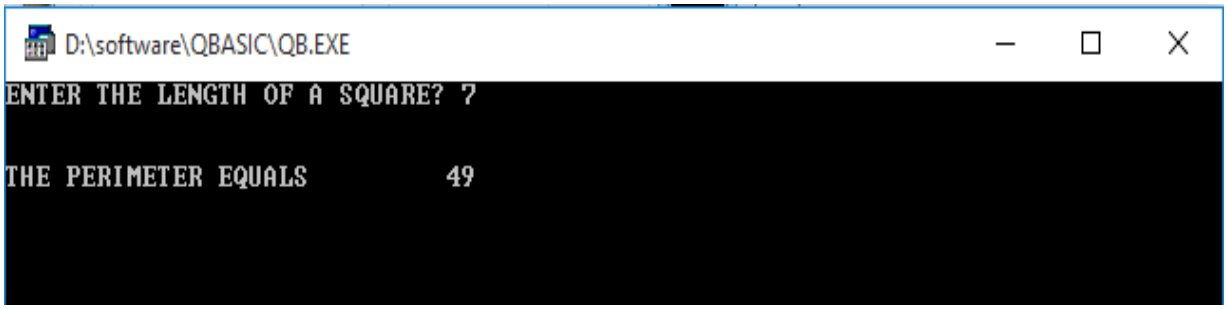

Figure 3: BASIC object program

DOI: 10.24214/jecet.B.9.1.01520.

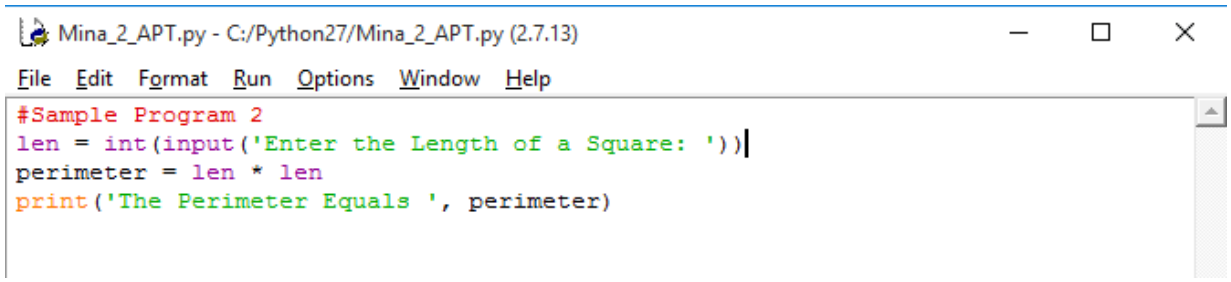

Figure 4: Python source program – Code Fragment

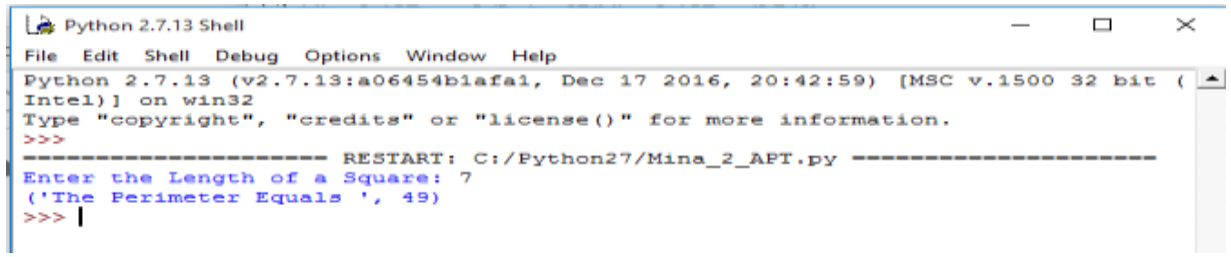

Figure 5: Python object program

5. CONCLUSION

Every programming language has its compiler, which is the translation tool for program execution according to lexical, syntactic and semantic structure of the underlying notations of the programming language in use. But, the functional logic of any computer program particularly specialized applications as user's program depends largely on its implementation algorithm or building block. Beyond the common criteria for selecting a particular language in teaching introductory computer programming; market demand, coverage and library extension are added advantages in choosing Python for teaching and learning computer programming at entry level.

REFERENCES

1. A. Ali and D. Smith, Teaching an Introductory Programming Language in a General Education course. *Journal of Information Technology Education and Innovations in Practice, 2014*, 13, 57-67. Retrieved from http://www.jite.org/documents/vol13/jitev13/pp057-067.pdf
2. ACE, *Introduction to C++ Programming*. Creative Common Attributions, 2007, Retrieved from scholar.google.com on 24th August, 2019.
3. D. Beyon, the 'Language' of Informatics: The Nature of Information Management. *International Journal of Technical Computing, 2009*, 9(2), 92–102.
4. S.S. Ghanzala and I. Noman, A Qualitative Study of Major Programming Languages to Computer Science Students. *Journal of Info and Communication Tech, 2016,*. 10, 1, 24-34.

5. R.I. Halterman, *Learning to Program with Python*.2011, Retrieved online from http://www.amazon.com on Monday 18th November, 2019.
6. B. Heinold, *A Practical Introduction to Python Programming,2012,* Retrieved from www.developersnetwork.com on 30th September, 2019.
7. T. Ioana and M. Chain, Analysis of Student Programming Errors in Java Programming Courses. *Journal of Emerging Trends in Computing Sciences*, 2012, 3(5), 739-742.
8. K. Kulli, P. Margins, L. Ali, T. Eno, The Role of Programming Experience in ICT Students' Learning Motivation and Academic Achievement. *International Journal of Information and Educational Technology, 2016,* 6(5), 331-398.
9. R. Lafore, *Data Structure and Algorithms in Java*. United State of America: Waite Group Press International,2007.
10. A.N. Ojekudo, Computer Programming Bridge. Port Harcourt: Emmanest Ventures and Data Communication,2019,
11. M. Poole and C. Whyley, The Compilers.2002, Retrieved from www.compsci.swan.ac.uk /cschris /compilers
12. T. Wu, Introduct ion to Object Oriented Programming with Java and Class Library. United State of America: Prentice Hall Publishing Co. 2010.

*** Corresponding Author: Douglas, T. Minafa-a,**

Department of Computer Science, Faculty of Natural and Applied Sciences, Post Graduate School, Ignatius Ajuru University of Education, Nigeria.