RESEARCH REVIEW JOURNALS
Volume-04
Issue-04
April-2019

ISSN: 2455-3085 (Online)
RESEARCH REVIEW International Journal of Multidisciplinary
www.rrjournals.com[UGC Listed Journal]

A Theoretical Review to Present the Comparison between Programming Languages

[1]Mrs. Manjiri A. Paranjpe and [2]Prof. Dr. Dhanajay T. Mandlik

[1]*Research Scholar, SIOM, SPPU, Pune*
[2]*Research Guide, SIOM, SPPU, Pune*

ARTICLE DETAILS

Article History
Published Online: 15April2019

Keywords
Language, Programming, software, factors.

ABSTRACT

Programming languages are utilized for managing the actions of computer devices. Many programming languages exist and brand new are being produced constantly. These programming languages start to be common with use to various programmers since there's surely a tradeoff between ease of use and learning, power and effectiveness of expression. Comparison of programming languages is a very common subject of debate among software engineers. Several programming languages are intended, specified, as well as implemented each year to be able to match the changing programming paradigms, hardware evolution, etc. Within this paper we show a comparative analysis between various programming languages.

1. Introduction

Programming languages are interesting and fascinating area of study. Personal computer scientists are likely to produce brand new programming language. 1000 various languages have been developed within the last couple of years. A number of languages appreciate large recognition & others bring in brand new features. Each language has its own drawbacks as well as benefits. A programming language is actually an artificial language developed to express computations that may be done by a machine, in particular a computer. Programming languages could be utilized to produce applications which manage the actions of a machine, to express algorithms exactly, or perhaps as a method of human interaction. Many programming languages explain computation in an imperative style, i.e., as a sequence of commands, though several languages, like the ones that support purposeful programming or maybe logic programming, begin using alternate forms of description.

The existing labor offers a comparison of different properties, paradigms, and features employed by a few of popular programming languages. With this particular assortment of languages and the widespread use of theirs, programmers and software designer must to be concerned about the advantages as well as drawbacks each language can provide to the software solution of theirs and be cautious whenever they make logical choices. These languages are actually compared within the qualities of reusability, reliability, portability, accessibility of equipment and compilers, efficiency, readability, expressiveness and familiarity. Various other criteria like the programming effort, run time efficiency, memory usage, and database connectivity are actually disclosed by applying as well as operating the identical set of applications using all of the languages under study.

Focus on programming languages

Even though programming languages are not always what we consider when we discuss software engineering trends, we chose them because of this very first experiment for a number of good reasons, like the following:

- They are essential artifacts in the story of software engineering.
- They stand for a unity of common attributes and purpose throughout many years of evolution.
- They provide a broad range of characteristics and also a long historical context, thereby affording us accurate analysis.
- The past of theirs is pretty well documented, and their essential qualities fairly well understood.

2. Factors of programming languages

Intrinsic factors

Intrinsic factors are the ones we are able to utilize to describe programming languages' overall style criteria. We have determined eleven this kind of factors:

- ***Generality:***Staying away from wonderful situations in the accessibility or maybe use of constructs and combining directly associated constructs to an individual, much more basic one.
- ***Orthogonality:*** The capacity to blend language constructs in a few significant way so that the interaction of constructs, or maybe the context of use, doesn't result in unforeseen actions or arbitrary limitations.
- ***Reliability:*** The extent to which a language helps the layout as well as development of dependable programs.
- ***Maintainability:*** The extent to which a language promotes ease of system maintenance, including, along with some other things, system readability.
- ***Effectiveness:***The extent to which a language design facilitates the generation of effective programs. Users as well as translators should quickly identify constructs which have suddenly costly implementations.
- ***Simplicity:*** The simplicity of a language layout, which includes these kinds of measurable facets as the minimality of necessary principles as well as the integrity as well as consistency of the structures of its.
- ***Printer freedom:*** The extent to which the language semantics are actually identified independently of machine specific particulars. Great languages should

not determine the qualities of object devices or maybe operating systems.

- ***Implementability:*** The extent to which a language comprises capabilities that are actually understood and could be implemented economically.
- ***Extensibility:*** The extent to which a language has common mechanisms for people to include features.
- ***Expressiveness:*** The capacity to express complicated computations as well as complicated details structures in appealing, user-friendly means.
- ***Impact or influence:*** The extent to which a language has affected the layout as well as evolution of the discipline and various languages of language design on the whole.

We chose these factors for the normal significance of theirs, family member completeness, and relative orthogonality. We do not maintain the list of ours is possibly orthogonal or complete - only that it's sufficiently abundant to allow us to shoot significant factors of programming language evolution.

Extrinsic factors

While intrinsic factors mirror qualities of the language itself, extrinsic factors characterize the historical context in which the language has emerged as well as evolved; these factors evolve with time. We represent these by chronological sequences of values rather compared to individual values. We have identified 6 groups of extrinsic factors for this study:

- institutional support,
- industrial support,
- governmental support,
- organizational support,
- grassroots support, and
- technology assistance.

For instance, grassroots support reflects the quantity of support that the language is actually getting from practitioners regardless of institutional, organizational, or maybe governmental pressures. Certain issues include

- How a lot of people consider that the primary language of theirs?
- How a lot of individuals are aware of that language?
- How a lot of user groups are actually focused on the use, evolution, or maybe dissemination of this particular language?

3. Types of programming languages

Programming languages

Programming languages are actually grouped into 3 main categories: scientific, non scientific & object oriented programming languages.

Scientific programming language

Scientific programming languages are actually procedural languages. They're command driven or perhaps declaration focused languages. The basis idea of theirs is the printer state. For scientific programming language, PASCAL as well as FORTRAN will be worn illustration.

Non scientific programming language

Non-scientific programming languages are procedural languages. They're closer and similar to nearly all of human language. Fundamental and COBOL will be examined in this specific category.

Object oriented programming language

Object Oriented Programming (OOP) is actually a technique of developing as well as using software. OOP languages include not only language syntax and compiler, though a whole growth atmosphere. These include a major library of well designed, simple to utilize items. It's a completely new means of thinking about problem solving with computers. An OOP language must Module should support details abstraction. The assistance of modules contain the following 7 attributes.

- Support of polymorphism and dynamic binding
- Ability to extend existing classes
- All data are defined by abstract data type based modules
- Automatic memory management
- Multiple inheritances.

4. Comparison of Programming Languages

Many works have been completed in the region of comparisons of programming languages. Jason Voegele given an individual analysis of several typical programming languages Comparison between extremely object based Java and non object oriented by Benchmarking. Jason Voegele given a analytical comparison of 9 various languages titled Comparison as well as "evaluation of programming languages". The effort was meant to give really high level info regarding the respective languages that are Eiffel, Smalltalk, Ruby, Java, C#, C++, Python, Perl and Visual Basic. Table one below shows the evaluation as well as comparison of several programming languages. This particular work however make an effort to evaluate six distinct programming languages in 3 categories of scientific, non scientific & object oriented programming languages.

Table 1: Evaluation and Comparison of Programming Language

	Eiffel	Smalltalk	Ruby	Java	C#	C++	Python	Perl	Visual Basic
Object- Orientation	Pure	Pure	Pure	Hybrid	Hybrid	Hybrid / Multi-Paradigm	Hybrid		Partial Support
Static/ Dynamic Typing	Static	Dynamic	Dynamic	Static	Static	Static	Dynamic	Dynamic	Static
Generic Classes	Yes	N/A	N/A	No	No	Yes	N/A	N/A	No
Inheritance	Multiple	Single	Single	Single	Single	Multiple	Multiple	Multiple	None

			class, multiple "mixins"	class, multiple interfaces	class, multiple interface				
Feature Renaming	Yes	No	Yes	No	No	No	No	No	No
MethodOverloading	No	No	No	Yes	Yes	Yes	No	No	No
Operator Overloading	Yes	Yes?	Yes	No	Yes	Yes	Yes	Yes	No
Higher Order Functions	Agents (with version5)	Blocks	Blocks	No	No	No	Lambda Expressions	Yes (???)	No
Lexical Closures	Yes (inlineagents)	Yes (blocks)	Yes (blocks)	No	No	No	Yes (since2.1)	Yes	No
Garbage Collection	Mark and Sweep or Generational	Mark and Sweep orGenerational	Mark and Sweep	Mark and Sweep or Generational	Mark and Sweep or Generational	None	Reference Counting	Reference Counting	Reference Counting
Uniform Access	Yes	N/A	Yes	No	No	No	No	No	Yes
Class Variables / Methods	No	Yes	Yes	Yes	Yes	Yes	No	No	No
Reflection	Yes (as of version 5)	Yes	Yes	Yes	Yes	No	Yes	Yes?	No
Access Control	Selective Export	Protected Data, Public Methods	public, protected, private	public, protected, "package", private	public, protected, private, internal, protected internal	public, protected, private, "friends"	Name Mangling	None	public, private
Design by Contract	Yes	No	Add-on	No	No	No	No	No	No
Multithreading	Implementation- Dependent	Implementation- Dependent	Yes	Yes	Yes	Libraries	Yes	No	No
Regular Expression s	No	No	Built-in	Standard Library	Standard Library	No	Standard Library	Built- in	No
Pointer Arithmetic	No	No	No	No	Yes	Yes	No	No	No
Language Integration	C, C++,Java	C	C, C++, Java	C, some C++	All.NET Languages	C,Assembler	C, C++, Java	C, C++	C (via DCOM)
Built-In Security	No	No?	Yes	Yes	Yes	No	No?	Yes (perlse c)	No
CapersJonesLanguageLevel*	15	15	N/A	6	N/A	6	N/A	15	11

5. Conclusion

This particular effort provides a comparison of several programming languages. In order to conclude the write up it'snote worthy which there are lots of other programming languages which are many improvement from the old ones. As various programming languages are now being produced, 2 characteristics (runtime or mind space) are being regarded as most. In truth, probably the fastest aren't constantly the smallest, so tradeoffs between these 2 attributes that are vital are now being maintained.

The engineering community is growing incredibly with each passing season as well as weeks.As they're coming with brand new up-to-date handsets as well as tablets every other day as well as the opponent too has progress loaded with the marketplace to stand at probably the highest place. That is the region programmer's and web designers are actually in the huge need already since they've an excellent information of programming languages are actually readily available and each of them as well defined function.

References

[1] Odeh, Ayman. (2019). Analytical and Comparison Study of Main Web Programming Languages -ASP and PHP. 1517-1522. 10.18421/TEM84-58.

[2] K P, Naveen Reddy & .Y, Geyavalli& .D, Sujani& M, Rajesh. (2018). Comparison of Programming Languages: Review. 9. 113-122.

[3] Yalamachali, Geyavalli. (2018). Comparison of Programming Languages:Review. 9. 113-122.

[4] Sheikh, Ghazala& Islam, Noman. (2016). A qualitative study of major programming languages: teaching programming languages to computer science students. International Journal of Information and Communication Technology.

[5] Alomari, Zakaria&Halimi, Oualid&Sivaprasad, Kaushik&Pandit, Chitrang. (2015). Comparative Studies of Six Programming Languages, arXiv, April 2015

[6] Salih, Shadman. (2014). Selection of computer programming languages for developing distributed systems. 10.13140/2.1.4424.6728.

[7] Stein, Martin & Geyer-Schulz, Andreas. (2013). A Comparison of Five Programming Languages in a Graph Clustering Scenario. Journal of Universal Computer Science. 19. 428-456.

[8] VujosevicJanicic, Milena &Tošić, Dušan. (2008). The role of programming paradigms in the first programming courses. The Teaching of Mathematics. 11.

[9] Fourment, Mathieu &Gillings, Michael. (2008). A comparison of common programming languages used in bioinformatics. BMC bioinformatics. 9. 82. 10.1186/1471-2105-9-82.

