

Emerged and emerging programming languages

Janyavula Mohan Phani Kumar¹, Shabad Prashanth², Barma Lohith³, Ananthula Ananth Sai Durga Hari Ram Prasad⁴

¹⁻⁴Department of Computer Science and Engineering, Lovely professional university, Phagwara, Punjab, India

ABSTRACT: In this file we are going to discuss about some of the emerged and emerging programming languages. We have provided the information about few programming languages which helps you to acquire knowledge. Acquiring knowledge is not about having bulk amount of information, it's all about providing the important and valuable content to the reader in an easy way. The information we gathered by the research is very precisely provided below. Which helps you to give all the subject content related to the programming language. We have written and modified the subject of information thoroughly which helps you accurately to gain knowledge in simple way. We have given some reference links down the page. For any queries you can refer them and gather much more information deeply and learn new things vastly.

Key Words: Object Oriented Programming, Encapsulation, Inheritance, Abstraction, Polymorphism.

1. INTRODUCTION

1.1 C

Dennis Ritchie created the C programming language at AT&T Bell Laboratories in 1972. C programming is regarded as the foundation for programming languages, which is why it is referred to as the "mother of programming languages." It is simple to learn and produces effective programmes. C was first used for system creation work, specifically the applications that compose the operating system [1]. The C language has a large library and many built-in features. C is a structured.

programming language that allows for the division of a complex program into smaller programmes known as functions. It also allows for the free flow of data between these functions.

1.2 C++

C++ was developed as an extension to the C programming language by Bjarne Stroustrup in 1979. C++ offers Object-Oriented Programming, which C does not provide [2]. C++ supports the five basic OOP features:

- ENCAPSULATION
- POLYMORPHISM
- ABSTRACTION
- DATA HIDING
- INHERITANCE

C++ has greatly influenced many programming languages, including C#, Java, and even recent versions of C. There are currently over 35 distinct operators in C++, ranging from arithmetic and bit manipulation to logical operations, comparisons, and more [3].

1.3 C#

C# is pronounced "C-Sharp." It's a Microsoft-developed object-oriented programming language that runs on the .NET Framework. C# is related to other common languages such as C++ and Java, and it has origins in the C family. In the year 2002, the first version was released. C# 8, the most recent edition, was released in September of this year [4].

1.4 Python

Python is a well-known programming language. Guido van Rossum created it, and it was released in 1991. The name of this language was inspired by the British comedy series "Monty Python's Flying Circus," according to its creator, Guido Van Rossum [5]. The comedy was broadcast on BBC during the 1970s, and it provided some entertainment for the creator during the language's development. Van Rossum also desired a name that was both short and mysterious. Something that will capture everyone's interest. Python is written in the C programming language (actually the default implementation is

called CPython). Python is written entirely in English. However, there are several approaches: PyPy is a Python programming language (written in Python) [6].

1.5 Java

In 1995, James Gosling, labelled the "Father of Java," created Java. The project was started in the early 1990s by James Gosling and his team members. Java is currently used in internet programming, mobile devices, games, e-business solutions, and so on. The old name of JAVA is OAK. Oak is an extinct programming language developed by James Gosling in 1989 for Sun Microsystems' set-top box project. Later on, the language was renamed Java. Gosling chose the name Oak after an oak tree that stood outside his office. Java was originally designed for interactive television, but it was considered too advanced for the digital cable television industry at the time [7]. Sun Microsystems created the first Java compiler, which was written in C and used C++ libraries. The Java compiler is now written in Java, whereas the JRE is written in C.

1.6 Kotlin

Kotlin is a statically typed open-source programming language for the JVM, Android, JavaScript, and Native platforms. JetBrains is the company that produced it. The project began in 2010 and has always been open source. In February 2016, the first official 1.0 release was made. The most recent update is 1.4.31, which was released on February 25, 2021. Both object-oriented and functional constructs are available in Kotlin. It can be used in both OO and FP forms, or a combination of the two. If you are doing or exploring functional programming, Kotlin is a great option. It has first-class support for features like higher-order functions, function types, and lambdas [8]. Kotlin is a programming language that is primarily used in Android development (where it is officially supported by Google and functions as an alternative to Java). However, because of its simplicity, it is used for other purpose also.

1.7 Java Script

JavaScript is a scripting language that is lightweight. It is intended for the development of network-centric applications. It works in tandem with and complements Java. Since it is combined with HTML, JavaScript is very simple to implement. It is cross-platform and open to all. It is a high-level object-oriented scripting language that's used to issue commands in a run-time environment. It is an interpreted language, which means that instructions are interpreted line by line rather than being compiled step by step. JavaScript becomes very complex as a result of this [9].

1.8 Go

Go is an open-source programming language that is also known as Golang. It creates compiled machine code binaries and is statically typed. When it comes to syntax, developers suggest Google's Go language is the C for the twenty-first century. This new programming language, on the other hand, provides tooling for securely using memory, handling objects, gathering garbage, and providing static typing in addition to concurrency [10]. Garbage collection, type protection, dynamic typing, and many specialised built-in styles including variable length arrays and key-value maps are all included. It also comes with a wide regular library. For effective dependency management, programmes are built using bundles. To create executable binaries, Go programming implementations use a conventional compile and connect model [11].

1.9 Fortran

Fortran could be a creating by mental acts language that is extensively employed in numerical, scientific computing, whereas out with the scientific community, algebraic language has declined in quality over the years, it still contains a robust user base with scientific programmers and is additionally employed in organisations like weather forecasters, money mercantilism, and in engineering simulations. algebraic language programs will be extremely optimised to run-on superior computers, and generally the language is suited to manufacturing code wherever performance is very important. Fortran could be a compiled language, or a lot of specifically it's compiled ahead-of-time. In different words, you need to perform a special step known as compilation of your written code before you're ready to run it on a laptop. this can be wherever algebraic language differs to understood languages like Python associated R that run through an interpreter that executes the directions directly, however at the value of calculate speed [12].

1.10 Julia

Julia could be a high-level, superior, dynamic artificial language. whereas it's a all-purpose language and may be wont to write any application, several of its options square measure well matched for numerical analysis and process science. Distinctive aspects of Julia's style embrace a sort system with constant quantity polymorphism during a dynamic

programming language, with multiple dispatch as its core programming paradigm. Julia supports coincident, (composable) parallel and distributed computing (with or while not victimization MPI and/or the intrinsic resembling "OpenMP-style" threads), and direct line of work of C and FORTRAN libraries while not glue code. Julia uses a just-in-time (JIT) compiler that's named as "just-ahead-of-time" (JAOT) within the Julia community, as Julia compiles all code (by default) to code before running it. Julia is garbage-collected, employs willing analysis, and comes with cost-effective libraries for floating-point calculations, arithmetic, random variation generation, and regular expression matching. several libraries square measure offered, together with some (e.g., for quick Fourier transforms) that were antecedently bundled with Julia and square measure currently separate. Many development tools support writing in Julia, like integrated development environments (e.g., Microsoft's Visual Studio Code, with extensions offered adding Julia support to day, e.g. providing debugging and linting support); with integrated tools, e.g. a profiler (and flame graph support offered for the intrinsic one), debugger, and therefore the Rebugger.jl package "supports repeated-execution debugging" and additional[13].

2. TABLE

Languages	Advantages	Disadvantages	Scope
C	Powerful and efficient language, portable language, built in functions	Concept of OOPs, run time checking, low level of abstraction	Operating system, Embedded systems, GUI, compiler design,
C++	Compatibility with C, object oriented, memory management, large community support.	Use of pointers, security issues, absence of garbage collector, absence of built-in threads.	GUI, operating system, cloud computing, DBMS, libraries.
C#	Object oriented language, robust memory backup, fast coding of backend patterns.	Poor cross platform UI, less flexible, low level concurrency.	Website development, game development, backend service, window application.
Python	Improved productivity, interpreted languages, dynamically typed, vast libraries support.	Slow speed, not memory efficient, database access, runtime errors.	Web and internet development, desktop GUI applications, software development, network programming.
Java	Platform independent, predictable and neutral, compact programs, object oriented.	Low speed, no backup facility, complex codes, requires memory space.	Mobile application, web application, desktop GUI application
Kotlin	Easy to learn, less bugs, easily maintainable, increases team productivity.	Slower complication speed, small developer community, fewer kotlin experts.	Android development, server-side development, web development, native development.
Java Script	Faster speed, ease of learning, reduction in server load.	Security issues, single inheritance, slow bitwise operation.	Web application, server application, mobile apps, games.

Go	Easy to learn and use, simple and clean syntax, security friendly, excellent documentation.	New and developing language, consumes more resources, development takes effort, less versatile	Package management, testing support, platform independent.
Fortran	Simple language, efficient compilers, dominating language of supercomputers.	Lack of dynamic storage, no user defined data structures, no satisfactory open-source compiler.	Useful in writing large and complex applications.
Julia	Directly executed as executable code, universally executable with languages like python, C, R, etc.	Packages are not mature or as well maintained, it takes too long to initially plot data.	Machine learning, risk management, 3D printer, specialized application in augmented reality(AR).

3. CONCLUSION

Python is emerging as the most common language for data science applications. The Python programming language is strongly supported by Facebook, Amazon Web Services, and, most notably, Google. Google introduced the Python programming language in 2006 and has used it for a variety of applications and platforms since then. Python has additional advantages that accelerate its ascension to the top of data science software. It works well with most of the cloud and platform-as-a-service providers. It has the distinct advantage of ensuring large-scale success in data science and machine learning by enabling multiprocessing for parallel computing. Python can also be expanded with C/C++ modules. Python's most appealing feature is that anyone interested in learning it can do so quickly and easily. Python, as opposed to other languages, supports a shorter learning curve and outperforms others by encouraging an easy-to-understand syntax.

REFERENCE

- [1] Stephen G. Kochan, .Programming in C 3rd edition. pg.1-10
- [2] Ulla Kirch-Prinz & Peter Prinz, A Complete Guide to Programming in C++. pg.3-9
- [3] B. Stroustrup & Addison-Wesley, The Design and Evolution of C++, 1994
- [4] Svetlin Nakov, Fundamentals of computer programming with C# . pg.19-22
- [5] David Amos, Dan Bader, Joanna Jablonski, Fletcher Heisler, Python Basics: A Practical Introduction. pg.13-19
- [6] Tim Hall and J-P Stacey, Python 3 for Absolute Beginners. pg.6
- [7] EckelPresident, MindView Inc, Thinking in Java Second Edition. pg.9
- [8] Stephen Samuel & Stefan Bocutiu, Programming Kotlin. pg.7
- [9] Dr. Axel Rauschmayer 2021, JavaScript for impatient programmers(ES2021 edition). pg.19
- [10] Alan A. A. Donovan & Brian W. Kernighan, The Go Programming Language. pg.11-13
- [11] Viktor K. Decyk, Charles D. Norton, and Henry J. Gardner {research paper}, Why go? pg.2.
- [12] Stephen J. Chapmanpage, Fortran for Scientists and Engineers Fourth Edition. pg.22
- [13] Changhyun Kwon, Julia Programming for Operations Research, A Primer on Computing. pg.1-2.