

THE UNDERSTANDING PROGRAMMING LANGUAGES IN PROGRAMMING ALGORITHMS

Rahmawati Dewi¹, Nery Adelia Br Purba², Tia Dwi Kartika³, Sri Meitiara⁴, Sotar Ferdinan Hutabarat⁵

^{1,2,3,4,5} Sistem Informasi STMIK Kaputama Binjai, Medan, Indonesia

Email: ¹rahmazahra202@gmail.com, ²neryadelia373@gmail.com, ³tiadwikartika@gmail.com, ⁴tiaragintingg.07@gmail.com, ⁵sotarferdinanhutabarat@gmail.com

*Corresponding email: neryadelia373@gmail.com

ABSTRACT - *A programming algorithm is a coherent progression that is followed to address a specific problem or task. A programming language is a guide used to convey tasks to a computer. Programming languages include: C, C++, Java, Python, JavaScript. Each programming language has its own advantages and disadvantages. This article is expected to provide a broad understanding of the programming languages used in programming algorithms. The method used is a literature study which presents a programming language that is easiest to learn for beginners.*

Keywords: Programming algorithm, Programming languages

INTRODUCTION

An algorithm, as defined by Goodman, Hedet, and Niemi, is a finite sequence of clearly defined operations, each of which requires a limited amount of memory and time to solve a problem. Sunarto defines a program as a collection of instructions expressed in a language, code, scheme, or other format. When these instructions are combined with a medium that can be read by a computer, they enable the computer to perform specific functions, such as preparing instructions for design purposes.

From the definitions of algorithms and programs (programming), it can be concluded that an algorithm represents deliberate steps, whereas writing a computer program is the act of creating a program. A programming algorithm consists of systematic and logical steps used to develop computer programs by utilizing programming languages to solve problems and achieve specific goals (Samsudin, 2020).

A programming language is a language used by software users to communicate with computers (Mewati Ayub, 2023). Programming languages are essential tools in the development of programming. This article discusses several programming languages commonly used in algorithmic computation, including C, C++, Java, Python, and JavaScript. The focus is on understanding the advantages and disadvantages of each programming language.

Computers are one of the tools frequently used to facilitate human life. They play a vital role in education and the workplace. Beyond writing articles or creating engaging presentations, computers can be used for more advanced purposes. For example, computers can access the internet and connect people with one another. In the professional world, computers are used to create reports, organize data, edit advertisements, and even develop programs such as links or applications that can be utilized within an organization.

A computer can be instructed to create connections or applications through programming. Programming is a method or sequence of steps used to instruct a computer on how to operate. To provide commands or instructions to a computer, a medium is required to connect human intentions with the computer that receives those commands. Programming languages function as communication tools between humans and computers.

Basically, a computer is an electronic machine that operates properly by understanding low-level languages, particularly binary conditions such as 1 or 0, which represent the presence or absence of electrical voltage (Luqman Hakim, 2018).

Programming languages can be categorized into several types, such as low-level programming languages, which are closer to machine language, and high-level programming languages (such as Python, Java, and C++), which are easier for humans to understand and use. Although the fundamental concept of instructing computers to perform specific tasks remains the same, the syntax and writing rules of each programming language differ.

An algorithm is a technique that can be communicated as a finite sequence. It is also a collection of instructions designed to solve a problem. These instructions can be described step by step from beginning to end. Every problem has initial conditions that must be met before computation can take place. Computation also involves repetitive processes (iteration) and decision-making until the desired outcome is achieved (Maulana, 2017: 8).

This article discusses programming languages within the context of programming algorithms. The purpose of this article is to provide readers with an understanding of programming languages, the various types of programming languages, and the differences among them.

RESEARCH METHODOLOGY

In programming activities, there are several programming languages that can be used. A programming language serves as a communication framework used to provide instructions to a computer. Programming is the process of creating programs or activities related to program

development. A person who creates programs is called a programmer. To communicate with a computer, programs utilize programming languages. Algorithms play an important role in program development; therefore, some experts state that a program is the combination of an algorithm and a programming language (Yuniansyah, 2020).

The method used in this research is a literature study. The authors obtained the required information from reliable sources such as books and academic articles related to programming languages and algorithms. The study analyzes the similarities and differences among selected programming languages. C, C++, Java, Python, and JavaScript were chosen as the programming languages for this method. By collecting data on these programming languages, the study aims to identify the strengths and weaknesses of C, C++, Java, Python, and JavaScript.

RESULTS AND DISCUSSION

The term *algorithm* originates from the name Al-Khwarizmi, which is found in the work of Abu Ja'far Muhammad ibn Musa Al-Khwarizmi, a Persian mathematician, in his book entitled "*Al-Jabr wa al-Muqabala*." Al-Khwarizmi was born before 800 AD and died after 847 AD. His full name was Abu Abdullah Muhammad ibn Musa, and he became known as Al-Khwarizmi because he came from Khwarizm, a region located northeast of the Caspian Sea (Abdurrahman, 2013).

An algorithm is a fundamental concept that must be understood or mastered by anyone who intends to solve problems using computers through program development. Essentially, developing a program means creating a tool to address a specific problem. A program consists of a set of instructions given to a computer, while the strategy and proper sequence of steps within a program are referred to as an algorithm. Programs are created using programming languages. Therefore, it can be stated that a program is the execution of a programming language.

Programming languages are used to develop computer programs. Both system programming and application programming are carried out using programming languages. Based on their level of abstraction and ease of use, programming languages are generally classified into two categories: low-level programming languages and high-level programming languages. Low-level programming languages are more difficult to learn because their instructions consist of specific codes and symbols that are not easily understood by general

users. On the other hand, high-level programming languages are easier to learn because their instructions use terms derived from natural languages (such as English).

Machine language and assembly language are examples of low-level programming languages, whereas BASIC, FORTRAN, COBOL, PASCAL, C++, and DELPHI are examples of high-level programming languages. In addition to these, there are many other programming languages in existence. This article focuses on high-level programming languages, specifically C, C++, Java, Python, and JavaScript.

1. C Programming Language

The C programming language was designed by **Dennis M. Ritchie** in **Murray Hill, New Jersey, United States**, at **Bell Telephone Laboratories, Inc.**, in **1972**. Initially, the C programming language was used only within Bell Labs research facilities in **1978**. After the publication of the book *The C Programming Language* by **Dennis Ritchie** and **Brian Kernighan**, the C programming language gained widespread popularity. The emergence of the C language later became the foundation for several other programming languages such as **Java**, **C++**, **PHP**, and others (**Hidayat, 2018: 21–23**).

The following are some advantages of the C programming language:

- a) For beginners, it is recommended to learn procedural programming first, which contrasts with object-oriented programming. Therefore, mastering the C programming language is highly recommended, as it is based on procedural programming.
- b) C is a programming language that is widely used to develop various types of applications and is often referred to as a universal programming language, ranging from system software and applications to compilers for other programming languages.
- c) C code is versatile and easily adaptable to various computer platforms. Applications developed on the Windows operating system can be ported to Linux without significant modifications.
- d) The execution process of C programs is faster because the C programming language can interact directly with computer hardware.

2. C++ Programming Language

Bjarne Stroustrup and his team at Bell Laboratories (AT&T, USA) developed the C++ programming language to facilitate object-oriented programming in simulation projects. The earliest variant of C++ was known as “*C with Classes*.” According to Kirk-Prinz and Prinz (2002), the name C++ is derived from the C programming language, where the operator ++ represents an increment operation in C.

One of the advantages of the C++ programming language is its ability to run efficiently on various machines, ranging from personal computers to centralized servers. In addition, C++ can be used across many operating system environments and frameworks, such as DOS, UNIX, and others.

In general, documents in C++ programming can be divided into several types, namely header files, library files, source (editor) files, object files, and executable files (Anam et al., 2021). Header files are text files that are included in a program during the compilation process and contain step-by-step instructions for linking files through compiler directives. All header files have the .h extension.

Library files contain precompiled programs that provide basic functionalities used in C++ programming. These libraries are integrated and can be embedded into programs as needed. Source (editor) files are text files with the .cpp extension, typically created using development environments such as Dev-C++. Object files are files with object extensions produced during the compilation process. Meanwhile, executable files, which have the same name as the source file and can be executed directly from the DOS Prompt, are the final output of the linking process.

3. Java Programming Language

James Gosling and his colleagues at Sun Microsystems developed the Java programming language, which adopts several commands and syntactic elements from C++. Its initial development began in 1991 under the name “*Oak*,” and in 1995, Java was released for public use. As a programming language based on Object-Oriented Programming (OOP), Java incorporates key concepts such as encapsulation, which refers to the practice of wrapping data and methods within a class so that it is treated as a modular unit; inheritance, which allows data and methods to be passed from one class to another so that all features of the parent class are inherited by the child class; and polymorphism, which enables objects to take multiple forms by allowing methods with the same name to exhibit different behaviors during program execution (Hermawan, 2004).

Unlike C++, Java applies the principle of “Write Once, Run Anywhere.” Java programs are compiled into bytecode, enabling them to run on various platforms without modification. Another significant advantage of Java is its garbage collection mechanism, which automatically manages memory by freeing objects that are no longer in use. This feature eliminates the need for developers to explicitly deallocate memory, thereby reducing memory management errors and improving program reliability (Kadir, 2021).

4. Python Programming Language

Python was first created by Guido van Rossum at the Centrum Wiskunde and Informatica (CWI) in the Netherlands in the mid-1990s. The Python language was influenced by the ABC programming language. Python is known for its simplicity and readability, resembling the English language, and it does not require a compilation process. This allows Python scripts to be executed immediately after being saved without additional steps. Python is a powerful programming language that is widely used in application development across various fields (Pongsongga, 2021).

Compared to other programming languages, Python is relatively easy to learn and use due to its simple syntax, which is easy to read and remember. Python is a multifunctional language that supports various programming paradigms and application domains. Writing code in Python is generally more efficient than in languages such as C, C++, or Java. Additionally, Python can interoperate with other programming languages, as Python code can call C or C++ code and can also be called from other languages.

However, Python also has several limitations. Certain tasks cannot be handled efficiently by Python, as it is not as fast or as effective as statically compiled languages such as C. Python is also less suitable for portable application development, particularly for Android and iOS platforms. As an interpreted language, Python is not ideal for low-level execution tasks, making it difficult to develop high-performance 3D games with advanced graphics. Furthermore, Python has limitations in accessing certain database systems and cannot be used as a low-level execution language for specific system components. Instead, it functions more effectively as a front-end scripting interface. Overall, Python offers a trade-off between productivity and flexibility (Wardana, 2019).

5. JavaScript Programming Language

JavaScript (JS) is a programming language that can be applied to create websites, games, and applications, and it is widely used by developers. JavaScript has 1,444,231 types of libraries, and this number is expected to continue increasing in the long term. In addition, major technology organizations such as Microsoft, Uber, Google, Netflix, and Meta use JavaScript in some of their tasks. The JavaScript programming language has several specific advantages, including full integration with HTML and CSS, the ability to solve simple problems easily, and support for most internet browsers while being naturally dynamic. JavaScript is a major programming innovation that is able to consolidate these three focuses; therefore, it is often used as a tool for creating program interfaces (Wali, 2023).

JavaScript also provides several benefits. When running system code, there is no need to use a special compiler because internet browsers can execute the code automatically. JavaScript script code is easy to learn for beginner software developers, and error handling is relatively simple, allowing issues to be fixed efficiently. JavaScript is multi-platform in nature, as it can run on various frameworks, making it easy to develop applications according to client needs and technological advancements. In addition, JavaScript requires relatively small memory resources, resulting in fast execution time, lightweight performance, and easy-to-use interfaces. Furthermore, JavaScript has very broad and stable local community support. However, JavaScript also has several limitations. It is easy to exploit because it is not difficult to create and understand for web designers, it has the potential to be misused, such as for monitoring various digital activities, and it contains many inline codes, which makes code obfuscation difficult.

CONCLUSION

To create a program, a programming language is required. The programming language used is chosen based on the needs and the level of comfort of the programmer. In this article, programming languages are categorized based on their level of ease. Python is considered one of the easiest programming languages to learn and use because it has simple and readable syntax, making it suitable for beginners. In addition, the large number of Python libraries and extensive documentation make it easier to learn and develop applications. JavaScript is also relatively easy to learn, especially for those who are already familiar with HTML and CSS. JavaScript is commonly used for web development and has many accessible resources and strong local community support. The C programming language is a low-level language; although it has a basic syntax structure, it requires a deeper understanding of programming concepts such as pointers and memory allocation. C++ is more complex than C, and although it has a syntax structure similar to C, it also introduces object-oriented programming concepts that may be misunderstood by beginners. Java is also included in the category of more complex programming languages, as it has more complicated syntax and requires a deeper understanding of object-oriented programming concepts. It should be noted that each programming language has its own specific characteristics; therefore, the choice of programming language depends on the needs and abilities of the programmer. For beginners, using the Python programming language is recommended because it is simpler and easier to learn.

REFERENCES

Anam, Syaiful, dkk. 2021. Cara Mudah Belajar Bahasa Pemrograman C++. Malang: Universitas Brawijaya Press.

Abdurrahman, M. Y. 2013. Cara-Cara Belajar Ilmuan Muslim Pencetus Sains-Sains Canggih Modern. Yogyakarta: Diva Pess.

Ayub, Mewati. 2023. Belajar Bahasa Pemrograman Python Dengan Visualisasi. Yogyakarta : Zahir Publishing.

Hakim, Luqman. 2018. Bahasa Pemrograman. Yogyakarta : Deepublish.

Hermawan, B. 2004. Menguasai JAVA 2 & Object Oriented Programming, 1e. Yogyakarta: Andi Offset.

Hidayat, Reza Akbar. (2018).Ajarincode : Aplikasi Pembelajaran Bahasa Pemrograman Berbasis Web. Jurnal Sains Terapan Teknologi Informasi, Vol. 10(2), hal.21-23.

Kadir, Abdul. 2021. Logika Pemrograman Menggunakan Java. Jakarta : PT Alex Media Komputindo.

Maulana,G,G. (2017). Pembelajaran Dasar Algoritma Dan Pemrograman Menggunakan El-Goritma Berbasis Web. Jurnal Teknik Mesin, Vol. 06, hal.8.

PONGSONGGA, T. (2021).ANALISIS DAN PERBANDINGAN KINERJA RESTFUL API PADA BAHASA PEMROGRAMAN PHP 5, PHP 7, PYTHON v3. 7 DAN PYTHON v3. 8(Doctoral dissertation, Universitas Hasanuddin).

Prinz,Ulla Kirch dan Prinz Peter.2002. A Complete Guide to Programming in C++. Amerika: Jones and Bartlett Learning.

Samsudin, dkk. (2020). Perancangan Sistem Informasi Pembelajaran Algoritma dan Pemrograman Berbasis Web pada Program Studi Teknik Informatika STMIK ERESHA. Jurnal Informatika Universitas Pamulang, Vol. 5(4), hal. 521-528.

Wali, Muhammad. 2023. Pengantar 15 Bahasa Pemrograman Terbaik Di MasaDepan (Referensi & Coding Untuk Pemula). Jambi : PT Sonpedia Publishing Indonesia.

Wardana. 2019. Belajar Pemrograman dan Hacking Menggunakan Python. Jakarta : PT Alex Media Komputindo.

Yuniansyah. 2020. Algoritma dan Pemrograman Menggunakan Bahasa Pemrograman Jawa (Teori dan Aplikasinya). Bogor : Penerbit Lindan Bestan.