PYTHON AND C++ ARE PROGRAMMING LANGUAGES DESIGNED FOR OOP

Yigitaliev Ruzimatjon,
Teacher of the Department of Digital Technologies and Mathematics of Kokand University

Annotation: This article will explain features, advantages and key differences between Python vs C++ in detail.

Key words: OOP (Object Oriented Programming), C++, Python, syntax, programming language, function, attributes, advantages.

Python and C++ are programming languages designed for OOP

Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

OOP focuses on the objects that developers want to manipulate rather than the logic required to manipulate them. This approach to programming is well-suited for programs that are large, complex and actively updated or maintained. This includes programs for manufacturing and design, as well as mobile applications; for example, OOP can be used for manufacturing system simulation software.

Python and C++ are two different languages that have different features and different behavior. Both these languages have one thing in common i.e. strong support for object-oriented programming.

C++ Features

- Compiled language
- Strongly typed, case sensitive language.
- Machine independent or portable and modular.
- Fast and efficient
- Syntax based, powerful
- Uses pointers and has a huge functions library.
- The object-oriented programming language. It supports the following OOP features:
- Classes and objects
- Abstraction
- Encapsulation
- Polymorphism
- Inheritance

Python Features

- It is easy to learn and has clear syntax.
- It is extensible to a greater degree.
- Python is free, open-source, and cross-platform.
- It is an object-oriented programming language with high readability and reliability.
- Can be used for prototyping and testing of code that can later be used to develop a full-fledged application using the other higher-level languages.

- Ships with a huge standard library consisting of XML parsers excel interface etc.

Table of Differences Between Python Vs C++

Comparison Parameter	C++	Python
Compilation	Compiled	Interpreted
Usage	Not easy to write code.	Easier to write code.
Nature of language	Statically typed	Dynamically typed
Portability	Not portable	Portable
Garbage collection	Does not support Garbage Collection.	Supports Garbage Collection.
Installation	No difficulty	Difficult to install
Types	Data types bound to names checked at compile time.	Bound to values, checked at runtime.
Scope of the variables	Limited within the loops or blocks.	Accessible outside the loops or blocks.
Rapid prototyping	Not possible	Possible
Functions	Restrictions on the type of parameters or return value.	No restrictions on the type of parameters or return value.
Efficiency	Difficult to maintain.	Easier to maintain
Syntax complexity	Uses blocks and semicolons.	No blocks or semicolons.
Speed of execution	Faster	Slower
Performance	High performance	Low performance
Popularity	More popular for embedded or enterprise applications.	Most popular for machine learning.
Simplicity and usability	Difficult to learn and is used in a low-level application.	Simple and is used for machine learning or web applications.

Key Differences Between C++ and Python

1) Compilation. C++ is a compiled language. C++ compiler generates an object code from the C++ source code and is then executed to produce the output.

Python is an interpreted language. The Python code with an extension py need not be compiled. We can directly pass it to the Python interpreter and generate the output.

2) Usage. C++ has a lot of features and also has a comparatively difficult syntax. It is not that simple to write the C++ code.

Python is easy to write and has a clear syntax. Hence writing Python programs is much easier when compared to C++.

3) Nature of Language. C++ is a statically typed language i.e. the declaration of a variable, the data type of variables, etc. are verified at compile time. This keeps the source code error-free at runtime.

Python, on the other hand, is not statically typed. There is no type checking done at compile time. Hence, the code is prone to errors.

4) Portability. C++ is not portable i.e. we need to recompile the code on every different platform. C++ is mainly “Write Once, Compile Anywhere”.

Python is portable. It is also cross-platform and we can execute programs on any platform.

5) Garbage Collection/Memory Management. In C++, memory management is manual. C++ does not support automatic garbage collection of resources.

Python, on the other hand, has a feature of automatic garbage collection. Its memory management is system-controlled.

6) Rapid Prototyping. We cannot do rapid prototyping using C++.

Using Python, we can do rapid prototyping of code so that it can be used later for building applications using higher-level languages.

7) Scope of the Variables. C++ has code demarked by blocks using curly braces ({}) and loops. The scope of the variables is limited to these blocks and loops demarked by {}.

The scope of the variables used in Python is not limited to blocks or loops. The variables are accessible even outside the curly braces.

8) Installation. C++ can be easily installed on Windows. Python, however, is difficult to install.

9) Types. In C++ data types are bound to names and get checked at compile time. This reduces the possibility of any errors at runtime.

In Python, the data types are bound to values and are checked at runtime. Code may be more error-prone at runtime as we do not catch those errors at compile time.

10) Functions. Functions are blocks of codes with one or more parameters and a return value. Each of the parameters and return value has a type.

In C++, the types of parameters and return type during function call has to match with that in the definition of the function.

In Python, there is no such restriction on parameter and return types.

11) Efficiency. C++ code is difficult to maintain as it can get complicated to read as solutions become bigger.

Python, on the other hand, has clean code and simple syntax. The source code for Python is easier to maintain.

12) Syntax Complexity. In C++ there is a clear demarcation of the code by using blocks enclosed in {}, semicolons indicating the end of the statement, etc. Thus in C++, the syntax is well organized.

In Python, there are no blocks or semicolons. Instead, Python uses indentation.

13) Speed of Execution. As far as speed of execution is concerned, C++ programs run faster. In fact, C++ is known and used widely in applications that are required to run faster like gaming platforms.

Python, on the other hand, runs slowly. Moreover, Python programs run slower than Java programs. Hence, we employ Python specifically for applications that can compromise on speed.

14) Performance. C++ is a statically typed language, thus we have fewer errors to take care of at runtime. C++ also creates a more robust and faster runtime code. This makes C++ a language with high performance.

Python being dynamic, has a possibility that some errors or unwanted situation may arise at runtime. So as far as performance is concerned, Python lags behind C++. But when it comes to machine learning, Python is the one that has the upper hand.

15) Popularity. Python is easy to learn and easy to put in practice when compared to C++ which becomes harder as we advance through its features. Another advantage of Python is its libraries that allow us to write any functionality especially data analysis and machine learning.

So popularity-wise Python scores over C++. Especially for the development of machine learning applications, it is the number one choice for programmers.

16) Simplicity and Usability. Python with its simplicity and easy to use features allows us to write concise, easily readable code, etc. This is helpful when we develop complex applications for machine learning as we don’t have to struggle with a programming language.

Secondly, Python is easy to learn and is a simple language. The same cannot be said about C++. C++ is more of a lower-level language that is easier for computers than humans.

Thus Python scores on these parameters especially when we have to choose between C++ and Python for developing machine learning applications.

Key Advantages of Python

- One of the major advantages of Python language is clean, simple, and straightforward syntax. For C/C++ programmers, the syntax seems familiar but easy without semicolons and braces.
- Python has a huge standard library that has CSV and zip file readers/writers, several XML parsers, a library for using every internet protocol and data type.
- The language is great for building web applications mainly because of its simplicity and efficiency.
- Python supports “duck typing” i.e. we can go ahead and call any object without worrying about its specific type etc.
- Especially suitable for machine learning development.

Advantages of C++ Over Python

- The major advantage of C++ is performance. C++ performs efficiently and the speed is faster when compared to Python.
- C++ is suitable for almost every platform including embedded systems whereas Python can be used only on certain platforms that support high-level languages.
- Being a strongly typed language, C++ is more predictable than Python which is dynamically typed. This feature also enhances the performance of C++.
- C++ can be used for system programming including writing Operating systems.
- We can also use C++ to learn low-level programming as the language is closer to hardware. With Python, such a feat is not possible.

Conclusion

C++ and Python are two different languages that have very diverse features as well as applications. While Python has easy syntax, high readability, etc. it is far behind C++ in terms of system programming, performance, and speed. While Python can be the best choice for machine learning development, C++ is best for a whole range of applications including system

programming as C++ offers us all the features available under the sun. In this article, we have seen the major differences between C++ and Python and the advantages of Python and C++ over Python as well.

References:

1. Davronjon, A., & Gulmiraxon, K. (2022). WIDESPREAD INTRODUCTION OF DIGITAL TECHNOLOGIES IN THE REAL SECTOR OF THE ECONOMY, AS WELL AS IN AGRICULTURE AND WATER MANAGEMENT. World Economics and Finance Bulletin, 9, 167-172.
2. Abdullajonov, D. S. O., & Kasimova, G. K. Q. (2022). DEVELOP A TRAINING PROGRAM FOR YOUNG PROFESSIONALS IN EDUCATIONAL INSTITUTIONS, WHICH IS THE CORE OF CYBERSECURITY. Academic research in educational sciences, 3(6), 185-192.
3. Shokirjon o‘g‘li, A. D. (2023). AXBOROT TEXNOLOGIYALARI SOHASIDA ILMIY IZLANISHLAR OLIB BORISH TENDENSIYALARI. QO ‘QON UNIVERSITETI XABARNOMASI, 1223-1227.
4. Abdullajonov, D., & Qosimova, G. (2022). OZBEKISTONDA KIBERXAVSIZLIK VA RAQAMLI IQTISODIYOT RIVOJLANISHINING AXBOROT JAMIYATI SHAKLLANISHIDAGI ORNI. Евразийский журнал математической теории и компьютерных наук, 2(13), 29-37.
5. Shokirjon o‘g‘li, A. D., & Solijon o‘g‘li, T. U. (2022). The Application of Digital Technologies to Enterprises and Organizations Will Help Reduce Social and Economic Costs. Eurasian Journal of Learning and Academic Teaching, 4, 131-140.
6. Abdullajonov, D. (2021). RAQAMLI TEXNOLOGIYALAR ORQALI YANGI O’ZBEKISTONNING IQTISODIYOTINI RIVOJLANTIRISH, RAQAMLI IQTISODIYOTNING ISTIQBOLLARI. Экономика и социум, (12-1 (91)), 28-33.
7. Абдуллажонов, Д. Ш., & Одилов, А. (2023). ОСНОВЫ ВЗАИМОДЕЙСТВИЯ PHP И БД. Results of National Scientific Research International Journal, 2(11), 229-241.
8. Olimov, I. (2023). DEVELOPMENT OF ARTIFICIAL INTELLIGENCE IN OUR COUNTRY ALONG WITH INTERNET TECHNOLOGY INNOVATIONS. Modern Scientific Research International Scientific Journal, 1(2), 210-216.
9. Nuritdinov, J. T. (2021). ABOUT THE MINKOWSKI DIFFERENCE OF SQUARES ON A PLANE. Scientific reports of Bukhara State University, 5(3), 13-29.
10. Mamatov, M. and Nuritdinov, J. (2020) Some Properties of the Sum and Geometric Differences of Minkowski. Journal of Applied Mathematics and Physics, 8, 2241-2255. doi: 10.4236/jamp.2020.810168.
11. Mamatov, M. S., & Nuritdinov, J. T. (2020). On some geometric properties of the difference and the sum of Minkowski. ISJ Theoretical & Applied Science, 06 (86), 601-610. Soi: http://s-o-i.org/1.1/TAS-06-86-110 Doi: https://dx.doi.org/10.15863/TAS.2020.06.86.110
12. Mamatov , M., Nuritdinov, J. and Esonov, E. (2021) “Differential games of fractional order with distributed parameters”, International Scientific Technical Journal "Problems of Control and Informatics", 66(4), pp. 38–47. doi: 10.34229/1028-0979-2021-4-4.

13. Nuritdinov J.T. Minkowski difference of cubes. Proceedings of International Conference on Mathematics and Mathematics Education (ICMME - 2022). Pamukkale University, Denizli, Turkey, 22-24 September 2022; 88-90.
14. Mo‘sajonovna, Ismoilova E’zozaxon, Tuxtasinov Maqsadjon Murodjon o‘g‘li, and Bahodir Xoshimovich Karimov. "MIKROKONTROLLERLAR TARIXI VA ULARNING BUGUNGI KUNDAGI AHAMIYATI." O'ZBEKISTONDA FANLARARO INNOVATSIYALAR VA ILMIY TADQIQOTLAR JURNALI 2.19 (2023): 477-479.
15. Maqsadjon Murodjon O’G’Li Tuxtasinov, Abdumannon Kodirjonovich Jumakulov, and Xatamjon Mo’Ydinovich Xoldarov. "TALABALARNING MUSIQA VA SAN’ATGA BO’LGAN QIZIQISHLARINI OSHIRISHDA ZAMONAVIY TEXNIK VA ELEKTRON VOSITALARDAN FOYDALANISH" Oriental Art and Culture, vol. 3, no. 2, 2022, pp. 354-359.
16. TUXTASINOV, MAQSADJON. Texnologik ta’lim Jarayonida Texnik Ijodkorlik Rivojlanishida Raqamli Texnologiyalar. 2023.