C vs PYTHON

In this article, I will compare the very popular C and Python programming languages from the past to the present. In my opinion, If we want to understand programming languages and compare them correctly, we should examine how to occur them. In that case, we can start to examine how to occur programming languages.

Let's look at the question "What is programming language?" Humans has dealt with many problems from past to present. As the difficulty of the problems increased, it became very difficult to solve them. Computers were invented to solve difficult problems faster. However, for computers and humans to communicate, there must be a language layer in between. This layer is called "Programming language".

To solve problems using a computer, it is necessary to present the problems in a format that the computer can understand. Programming languages are used for this. Programming languages are a form of coding that allows to develop programs for any platform without going too deep into the computer, without dealing with concepts such as 0 and 1. Programming languages are just like spoken languages, they have their own spelling and grammatical structure.

Did you know that the first programming language was developed in 1843 ? Ada Lovelace's machine notes contain the first algorithm written to be processed by a computer. It has made a great contribution to laying the foundations of computer programs that have taken over most of our lives today. Its purpose was to assist Charles Baggage with his Bernoulli number calculations. Although Ada Lovelace's machine algorithm was a great development, the 20th century was the year when programming languages developed the most. Let's look at the history of the C and Python programming languages, which are the subject of the article.

Firstly, Let's start with the C programming language, which is one of the oldest modern programming languages. C programming language, was developed in 1972 by Ken Thompson and Dennis Ritchie at AT&T laboratories. The language, which was not very common in the years it was derived, started to spread rapidly after the book "C Programming Language" published by Brian Kernighan and Dennis Ritchie. With the rapid spread of the C programming language, developers and institutions created different C languages using these codes. This situation started to put system administrators in a difficult situation. To solve the emerging problem, the US National Standards Institute ANSI took over the situation and introduced the ANSI C standardization in 1989. This is how the foundation of the modern C programming language, which we still use today, was laid.

Now, let's look at the history of the Python language, which is used by many programmers with pleasure and ease, especially in fields such as data analysis and machine learning, due to its increasing popularity. Python is a high-level programming language developed by Guido Van Rossum in 1991. In the late 1980s, Guido Van Rossum started working on a high-level programming language in Amsterdam, where he lived, so that anyone could code comfortably. Van Rossum was planning to make his language inspired by ABC with this program. Rossum, who had previously contributed to ABC's development, was generally satisfied with ABC's features, although he thought it had shortcomings.

Rossum named the ABC-like programming language he created as Python. Although today the name is thought to come from the python, a snake species, it actually comes from Rossum's favorite comedy show "Monty Python's Flying Circus" broadcast on the BBC.

Now that we understand programming languages, we can start comparing C and Python languages in real terms. The most important and fundamental difference between these two languages is that their programming paradigms are different. The C language has the Procedural Oriented Programming paradigm while the Python language has the Object Oriented model.

In procedural programming, the program is divided into small parts called procedure. Procedures, simply consist of a series of computational steps to be carried out. During a program's execution, any given procedure might be called at any point, including by other procedures or itself.

In object-oriented programming, the program is divided into small parts called objects. Objects contain data in the form of attributes and code in the form of methods. In object-oriented programming, computer programs are designed using the concept of objects that interact with the real world. Both of these programming paradigms have their advantages over each other. For example, Procedural Programming is faster than Object Oriented Programming, but Object Oriented Programming is better than Procedural Programming in terms of data security.

Now let's compare these two languages in terms of **syntax**. We can start the comparisons over a simple function. For example, let's start with the "print" function, which prints a text to the screen. A C program consists of various tokens and a token is either a keyword, an identifier, a constant, a string literal, or a symbol. For example:

printf("Hello, World! \n"); //print function

There is 5 token:

printf

(

"Hello, World! \n"

)

;

In a C program, the semicolon is a statement terminator. That is, each individual statement must be ended with a **semicolon**. It indicates the end of one logical entity.

Printf("hello World!"); This is true!

Printf("hello World!") This is false!

As for Python, it actually looks similar to C. But python does not contain semicolons.

print("Hello, World!") //print function

There is 5 token:

print

(

"Hello, World! \n"

)

Print("hello world"); this is false!

Let's come to the **Identifier** benchmark. An identifier is a name used to identify a keyword. In c, identifier is a name used to identify a variable, function, or any other user-defined item. A Python identifier is a name used to identify a variable, function, class, module or other object.

Identifier in c starts with a letter A to Z, a to z, or an underscore '_' followed by zero or more letters, underscores, and digits (0 to 9). C does not allow punctuation characters such as @, $, and % within identifiers. C is a case-sensitive programming language. Thus, Alican and alican are two different

identifiers in C. This situation is similar in python. However Python has different conventions . Here are naming conventions for Python identifiers:

- Class names start with an uppercase letter. All other identifiers start with a lowercase letter.
- Starting an identifier with a single leading underscore indicates that the identifier is private.
- Starting an identifier with two leading underscores indicates a strongly private identifier.
- If the identifier also ends with two trailing underscores, the identifier is a language-defined special name.

Programming languages has **reserved words**. These reserved words may not be used as constants or variables or any other identifier names.

This is list of C reserved words:

auto	else	long	switch
break	enum	register	typedef
case	extern	return	union
char	float	short	unsigned
const	for	signed	void
continue	goto	sizeof	volatile
default	if	static	while
do	int	struct	_Packed
double			

This is list of Python reserved words:

and	exec	not
assert	finally	or
break	for	pass
class	from	print
continue	global	raise
def	if	return
del	import	try
elif	in	while
else	is	with

except	lambda	yield

Let's take a look at **lines and indentation**. Python provides no braces to indicate blocks of code for class and flow control. Blocks of code are denoted by line indentation, which is rigidly enforced. The number of spaces in the indentation is variable, but all statements within the block must be indented the same amount. For example:

```
if True:
   print ("True")
else:                                   That’s true!
   print ("False")
```

```
if True:
print ("True")
else:                                   That’s false!
print ("False")
```

However, C must contain braces to indicate blocks of code for function and flow control. No indentation is needed in c language.

```
If(True){
Printf(“True”);                 That’s true!
}
If(True)
Printf(“True”);                 That’s false!
```

Let's take a look at **Multi-Line Statements.** Statements in Python typically end with a new line. Python does, however, allow the use of the line continuation character (\) to denote that the line should continue. For example:

```
total = item_one + \
          item_two + \
          item_three
```

In c there is no line continuation character. For example:

```
total = item_one +
        item_two +
        item_three;
```

Let’s come to **Comment Lines**. In python, a hash sign (#) that is not inside a string literal begins a comment. All characters after the # and up to the end of the physical line are part of the comment and the Python interpreter ignores them. For example:

First comment

print ("Hello, Python!") # second comment

This produces the following result:

Hello, Python!

In c,

/* and terminate with the characters */ as shown below

/* my first program in C */

You cannot have comments within comments and they do not occur within a string or character literals.

Now let's examine the **data types**. Data can be of most types. As a person of age with children as a world address and alphanumeric designs. Python has several types of standards that are used for the tools used in the superdata and the intended uses for each.

Python has five standard data types:

• Numbers

• Twine

• List

• Group

• Dictionary

Number data lists store values. Matches the Number you give to a value.

Strings in Python denote their design.

Lists collect a lot of data together in a single structure. A list contains comma-separated and enclosed parentheses. Lists are just as similar to arrays in C. The difference is that all items belonging to a list are given differently.

Tuple is another array data type similar to a list. A tuple consists of a comma-separated set of values. But inside the lists, bundles are paid.

Basics of lists and tuples: Lists can be imported in parentheses ([ ]), and properties that can be enclosed in tuples in parentheses ( ) cannot be updated. Tuples as read-only lists.

Python's dictionaries are a type of hash table. They are useful like associative arrays or hashes in Perl, and they consist of key-value pairs. Dictionary key is pretty much its Python type, generically or strings. Values can be any bound Python object. A basic concept among predictions in dictionaries. It is incorrect that items are "ragged"; They're just out of order.

C has most data types:

• Integer

• Character

• Sailor without sinking

• Double

• Invalid

The integer data type in C is used for the full-case non-valued case. Octal values, hexadecimal values and decimal values can be stored int data in C.

The character data type only allows drawing of a single character. The character is 1 in size. It is the most basic data type in C.

Floating data type is used for floating point files in C programming. Float in C is used for decimal and exponential sights. Decimal (floating point numbers) are used for announcement with single precision.

Data in C is used to rust in decimal pair type (rock's dotted owner numbers) with double precision. It has tools in C.

The v data type in C is used for the type of no value. It does not provide a result to its caller. They have no values and no transactions. It is used to represent nothing.

If we compare the **efficiency** of these two languages, of course, the C language is much faster than the Python language. The biggest reason for this is the differences in the translation of languages into machine language. Two different methods are used when translating languages into machine language. These are the "compiler" and "interpreter" methods. C language uses compiler. The compiler takes code snippets and translates them into machine language before they even start running. Interpreter, on the other hand, reads the code line by line while it is running, and translates it into machine language, and continues every translated work. While the code translated into machine language by the compiler works directly, the code in the python language used by the interpreter is translated into machine language line by line, leading to a waste of time.

Now let's come to the usage areas of these two languages.

Perhaps one of the most popular applications of Python is data science. Python libraries for data analysis and visualization have incredible power. Packages called NumPy and Pandas, which are used for data analysis, make a lot of work in this area very easy. It is possible to say that there are more package sources for data visualization. It is stated that Python makes many tasks easy for data visualization with this category, which includes many packages such as matplotlib, Seaborn, ggplot2 and Bokeh. Python is an important language for any software developer who wants to enter the field of Machine Learning, which is very important for the future. Machine Learning uses algorithms that are constantly improved based on input data that helps the system "learn". One of the most popular libraries used by developers worldwide for working with Python on Machine Learning is TensorFlow. TensorFlow is a free open source library developed by Google. This library is used for research and production at Google. The most popular Python libraries used for machine learning are known as Keras and PyTorch. Python is also used in fields such as Web Development, Image Processing, Game Development, Medicine and Pharmacology, Biology and Bioinformatics, Neuroscience and Psychology, Astronomy.

The C programming language is generally considered the language of choice for the development of portable applications. In addition, when it comes to high-level software, this programming language can be applied. When we need to give an example of software developed with the C language, we come across the unix operating system. This programming language, which was developed in 1972, is frequently preferred for the software of applications that offer advanced options. In addition, software languages such as C++, Java, JavaScript, JavaApplet, PHP, C#, which are frequently used in the game industry, were inspired by the C programming language. In other words, this programming language can be said to be the beginning of important software languages. As for what you can do with the C

programming language, you can write drivers with this language. You can produce graphics and game software. You can also code machines such as the washing machine calculator, which is considered an embedded system. You can also make a robot with this software language. In addition to these, the unix kernel, Microsoft Windows programs, operating system applications and a large part of the android operating system were created using the C programming language. You can also use the C programming language to create a new programming language. In short, this programming language is preferred in every field and at every stage of technology.

From a **learning curve** perspective, python has a simpler syntax than c. Also, implementing data structures is easier in python. Since it is necessary to manually control the memory management in the c programming language, the python language does this task itself. When we look at it from these perspectives, it is an easier language to learn than the c language with python language. But python is a language that supports object oriented. Considering the Object Oriented side, it may be more difficult to use than C. Because it is necessary to learn the Object Oriented principles well.

Sources:

- https://www.geeksforgeeks.org
- https://talentgrid.io/tr
- https://tr.wikipedia.org
- https://www.tutorialspoint.com