Spectrum Journal of Innovation, Reforms and Development
Volume 26, April - 2024 | ISSN (E): 2751-1731
Website: www.sjird.journalspark.org

# WORKING WITH ARRAYS AND LISTS IN PYTHON AND C++ PROGRAMMING LANGUAGES

Shamsiddinova Maftunabonu Ulug'bek qizi,
Student of Bukhara State University
shamsidinovamaftuna4@gmail.com

**Abstract**
This article examines the strengths and weaknesses of the C++ and Python programming languages, their efficiency and usability for working with lists and arrays. Principles for implementing convenient methods for working with lists are presented.

**Keywords**: C++, Python programming language, list, working with lists, string, efficient integration, GUI, C programming language.

## Introduction

The use of information technologies in the educational system plays an important role in the educational process, not only in the theoretical knowledge, but also in conducting practical training and preparing them as mature personnel in all aspects. Information technology in education is a broad normative concept, and every subject that is conducted requires the use of this technique and technology. Because the use of modern teaching techniques gives positive results. The application of modern information technologies in the educational system opens wide opportunities for the use of new teaching methods in the educational process. The use of information technologies in the educational system is mainly related to the creation of personal computers and information technology software.

## Methodology

The C++ programming language is a general-purpose programming language developed as an extension of the C programming language. C++ is designed to provide a combination of high-level abstractions as well as low-level memory manipulation features, making it suitable for both system-level programming and application-level programming. Below are the main features and uses of C++ language:

Object-Oriented Programming (OOP): C++ supports object-oriented programming, which allows developers to separate code into classes and objects. It includes features such as encapsulation, inheritance, and polymorphism.

Low-level manipulation: C++ provides features for low-level memory manipulation, including pointers and manual memory management. This makes it suitable for tasks that require fine-grained control over system resources.

High Performance: C++ is known for its efficiency and high performance. It allows direct memory management and provides features such as inline functions that contribute to faster execution.

System Programming: C++ is commonly used for system-level programming tasks such as operating system development, device drivers, and embedded systems programming. The ability to communicate directly with hardware and manage resources makes it ideal for these applications.

Game Development: C++ is widely used in the gaming industry to develop high performance games. Popular game engines such as Unreal Engine and Unity use C++, and many game studios use C++ to write game logic and performance-critical components.

Application Development: C++ is used to develop a variety of applications, including desktop applications, productivity software, and large enterprise applications. It is chosen when performance is an important factor.

Middleware Development: C++ is often used to create middleware that provides a bridge between different software applications. This includes components such as libraries, frameworks, and APIs that facilitate communication between software components.

Graphics and Multimedia: C++ is commonly used in graphics programming, especially for creating graphical user interfaces (GUIs) and working with graphics libraries. It is also used in multimedia applications for tasks such as audio and video processing.

Networking: C++ is used in network programming, including the development of network protocols, server applications, and network systems. Its low-level capabilities make it suitable for efficient network programming.

Artificial Intelligence (AI): C++ is used in artificial intelligence programming, especially in the performance-critical parts of AI systems. Machine learning libraries and frameworks often have components implemented in C++ for optimal performance. C++ is a versatile language with a wide range of applications that require a combination of high performance and system-level control. making it a popular choice for a variety of domains.

In C++, lists can be implemented using the Standard Template Library (STL) container std::list. std::list is a doubly linked list that provides constant insertion and removal of elements at the beginning and end, and constant removal of elements in the middle, given an iterator.

The Python programming language has an easy-to-read syntax and a large standard library that supports many common programming tasks, such as connecting to web servers, searching for text with regular expressions, and reading and modifying files. is a high-level programming language that can be easily extended by adding new modules implemented in a compiled language such as C or C++ that can be embedded into an application to provide an automated interface. Its lights are characterized by the fact that its lights are located in the view of its expandable libraries. Python is very powerful and can have libraries that help with list and time operations. Allows you to write long and complex codes shorter and more precisely. Python is a convenient, clear, and powerful object-oriented programming language for teaching and learning programming at both entry-level and advanced courses, enabling programmers to work faster and integrate systems more effectively. It can also be compared to Perl, Ruby, Scheme or Java. Using lists (ie lists, tuples, and other data types) in Python is very simple and powerful. Some features of Python programming language:

Code can be grouped into modules and packages.

The language supports raising and catching exceptions, which helps to handle errors more accurately.

Data types are strongly and dynamically introduced. Mixing incompatible types (for example, trying to add a string and a number) raises an exception, so errors are caught more quickly.

Python includes advanced programming features such as generators and list comprehensions.

Python's automatic memory management saves you from having to manually allocate and deallocate memory in your code.

The result.

Information about the efficiency of the Python programming language for working with lists can be seen in the following example:

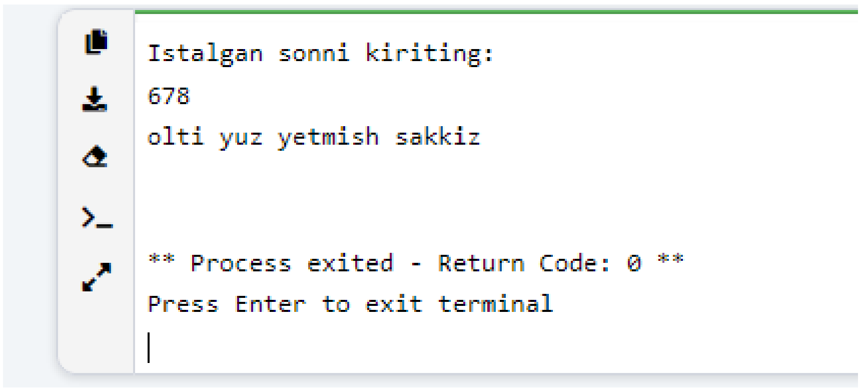
Example 1: Create a program that outputs a text display when a number in the range [0;10000] is entered?

Program code:

```
def entry(number):
    if count == 0:
        return "zero"
    if number>=10000:
        print("Enter numbers less than 10000!")
    if number<10000:
        units = ["", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine"]
        tens = ["", "ten", "twenty", "thirty", "forty", "fifty", "sixty", "seventy", "eighty", "ninety"]
        hundreds = ["", "one hundred", "two hundred", "three hundred", "four hundred", "five
hundred", "six hundred", "seven hundred", "eight hundred", "to 'nine hundred']
        thousands = ["", "one thousand", "two thousand", "three thousand", "four thousand", "five
thousand", "six thousand", "seven thousand", "eight thousand", "to 'nine thousand']
        unit_index = number % 10
        instant_index = (number // 10) % 10
        hundred_index = (number // 100) % 10
        thousands_index = (number // 1000) % 10
        unit = units[unit_index]
        onlik = onliks[onlik_index]
        hundreds = hundreds[hundred_index]
        thousands = thousands[thousand_index]
        form=thousands+ " "+ hundreds + " " + tens + " " + units
        shape=shape.strip()
        return form
number = int(input("Enter desired number: "))
form =script(number)
print(form)
```

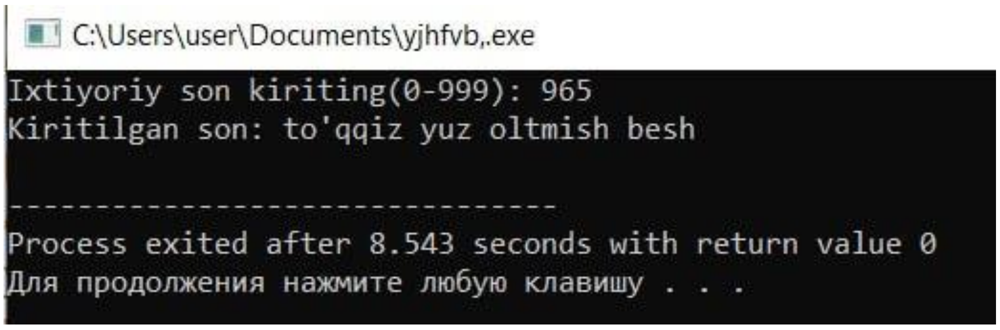
This code takes a parameterized function and returns the shape as a string by the shape name. In the code example above, lists are created for the units, tens, hundreds, and thousands of the entered number. It is used to divide the entered number into indices by numbers and find the unit of occurrence. unit_index variable is the unit of the entered number (module of the number). variable onlik_index represents the decimals of the entered number (the modulus of the result of division of the number by 10). hundredths_index variable represents the hundredths of the entered number (the modulus of the number divided by 100). The thousands_index variable represents the thousands of the entered number (the modulus of the number divided by 1000). Then, by assigning words to Uzbek numbers to the temporary variables, the work is done through the initial value (string) equal to the variable in the name of the form. To return, the string number format is returned as a collection of words. The example asks for a number to use the function with the console. It calls a function to convert this number to a number and output it to the console.

Program result:



```
Istalgan sonni kiriting:
678
olti yuz yetmish sakkiz


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```



```
C:\Users\user\Documents\yjhfvb,.exe
Ixtiyoriy son kiriting(0-999): 965
Kiritilgan son: to'qqiz yuz oltmish besh

--------------------------------
Process exited after 8.543 seconds with return value 0
Для продолжения нажмите любую клавишу . . .
```

Now let's solve this example in the C++ programming language:

Program code:

```
#include <iostream>
using namespace std;
string number(int number) {
string soz="";
   if(number==0){
      soz="zero";
   } else {
      string units[]={"", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine"};
      string decimals[]={"", "ten", "twenty", "thirty", "forty", "fifty", "sixty", "seventy", "eighty",
"ninety"  };
      string hundreds[]={"", "one hundred", "two hundred", "three hundred", "four hundred", "five
hundred", "six hundred", "seven hundred", "eight hundred"  , "nine hundred"};
      int unit=number%10;
      int tens=((number/10)%10);
      int hundreds=number/100;
      soz=hundreds[hundred]+" " + tenths[hundred] + " " + units[unit];
   }
   return set;
}
```

```
int main() {
    int n;
    cout<<"Enter arbitrary number (0-999): ";
    cin>>n;
    if(n>=0 && n<1000) {
        string string=numberstring(n);
        cout<<"Number entered: "<<word<< endl;
    }
  else if(n==1000){
    cout<<"thousand";
  }
  else {
    cout << "Error: The number entered is invalid or greater than 1000!"  << endl;
    }
    return 0;
}
```

Program result:

**Summary**

The easy-to-use Python and C++ programming languages have found widespread use in a variety of real-world applications. As these are high-level, dynamically typed, interpreted languages, they are expanding rapidly in the fields of error correction.

**References**

1. Bobojonova M.A. Python programming language: study guide//-Bukhara: Sadriddin Salim Bukhari, 2023. -108 p.
2. Rustamov H.Sh. Algorithmic languages and programming. Study guide // Bukhara: Durdona publishing house, 2022. 254 pages.
3. Jumayev J., Shamsiddinova M.U. Using Python's graphical capabilities in teaching the subject of definite integral// Pedagogical skill, 2023, No. 9, p. 240-245.
4. Jumayev J., Shamsutdinova M.U., Aslonov U.Sh. Use of Python capabilities in working with engineering drawings//Proceedings of the international scientific and practical conference on "Innovative solutions in industrial engineering". Bukhara, November 24-25, 2023. Pages 352-353.
5. Sharipov N. Z., Kuldosheva F. S., Jumaev J. Research of the Effect of Factors on the Process of Separation of Shadow Seeds from the Peel //Eurasian Research Bulletin. - 2022. - T. 7. – P. 86-91.