Aim:

Write a program to add, subtract, multiply and divide two complex number using **Operator Overloading** .

Theory:

Operator overloading is an important concept in C++. It is a type of polymorphism in which an operator is overloaded to give user defined meaning to it. Overloaded operator is used to perform operation on user-defined data type. For example, ‘+’ operator can be overloaded to perform addition on various data types, like for Integer, String(concatenation) etc. Almost any operator can be overloaded in C++. However, there are few operator which can not be overloaded. Operator that are not overloaded are follows :

- Scope Resolution Operator - `::`
- Member Access or Dot operator - `.`
- Pointer-to-member Operator - `*`
- Ternary or Conditional Operator - `?:`

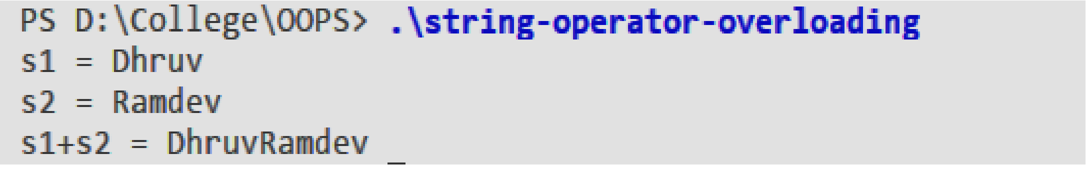
Syntax :

```
data_type classname :: operator symbol (arguments){
    //function body
}
```

Code :

```
#include <iostream>
#include <string.h>

using namespace std;

class String {
    int len ;
    char* str ;

    public :
    String () {
        len = 1 ;
        str = new char ;
        str[0] = '\0' ;

    }
    String (char* s) {
        len = strlen(s) ;
        str = new char[len + 1];
        strcpy( str , s );
    }
    void display(){
        cout << str << endl ;
    }
    String operator + ( String s ) {
        char* t = new char[len + s.len - 1] ;
        strcpy( t , str ) ;
        strcat( t , s.str ) ;
        return String(t) ;
    }

} ;

int main(){

   String s1 = "Dhruv" ;
   String s2 = "Ramdev" ;
   String s3 = s1+s2 ;
   cout << "s1 = " ; s1.display() ;
   cout << "s2 = " ; s2.display() ;
   cout << "s1+s2 = " ; s3.display() ;
    return 0 ;

}
```

Output :



```
PS D:\College\OOPS> .\string-operator-overloading
s1 = Dhruv
s2 = Ramdev
s1+s2 = DhruvRamdev _
```

Discussion :

In the above program we have created a class `String` to concatenate two strings by using operator overloading. We have overload `+` operator for adding two strings. First we created two objects of String
class and initialize two strings for two different objects and then we overload + operator or we can say that we invoke the operator function and then we created a string of third type of object which has
size of string one and string two and concatenate in it by using `strcpy()` function.

Learning Outcomes :

- We have learned that by using overloading operator our program will be more understandable. However, there are three methods to implement operator overloading that are: -
    - Member Function
    - Non-Member Function
    - Friend Function

- Operator overloading function can be a member function if the Left operand is an Object of that class, but if the Left operand is different, then Operator overloading function must be a non-member function. Operator overloading function can be made friend function if it needs access to the private and protected members of class.
- However, we cannot overload some of the operators that are given below: -
- Scope Resolution Operator - `::`
- Member Access or Dot operator - `.`
- Pointer-to-member Operator - `*`
- Ternary or Conditional Operator - `?:`