**Assignment No: 2**

**Problem Statement:** Implement a class Complex which represents the Complex Number data type. Implement the following operations: 1. Constructor (including a default constructor which creates the complex number 0+0i). 2. Overloaded operator+ to add two complex numbers. 3. Overloaded operator\* to multiply two complex numbers. 4. Overloaded << and >> to print and read Complex Numbers.

**Aim of Assignment:** Understand the concept of Operator overloading.

**Description:**

1. **Operator Overloading:** Operator overloading is a compile-time polymorphism in which the operator is overloaded to provide the special meaning to the user-defined data type. Operator overloading is used to overload or redefines most of the operators available in C++. It is used to perform the operation on the user-defined data type. For example, C++ provides the ability to add the variables of the user-defined data type that is applied to the built-in data types.
2. **Rules of overloading:** Existing operators can only be overloaded, but the new operators cannot be overloaded. The overloaded operator contains at least one operand of the user-defined data type. We cannot use friend function to overload certain operators. However, the member function can be used to overload those operators. When unary operators are overloaded through a member function take no explicit arguments, but, if they are overloaded by a friend function, takes one argument. When binary operators are overloaded through a member function takes one explicit argument, and if they are overloaded through a friend function takes two explicit arguments.
3. **Operator that cannot be overloaded:**

* Scope operator (::)
* Sizeof
* member selector(.)
* member pointer selector(\*)
* ternary operator(?:)

**OOP Concept Used:**

1. Class
2. Friend Function/Class
3. Overloading

**Conclusion:** In this experiment we have learnt regarding new concepts in OOP i.e. Operator Overloading