Assignment No. 01

Roll No. SC55 - Shreyas Chavhan

===================================================

# Complex Number

## Problem Statement -

Implement a class Complex which represents the Complex Number data type. Implement the following operations:

1. A 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 Number

## Outcomes -

To understand operator overloading concept.

## Theory -

### Operator Overloading:

You can redefine or overload most of the built-in operators available in

C++. Thus a programmer can use operators with user-defined types as well.

Overloaded operators are functions with special names the keyword operator followed by the

symbol for the operator being defined. Like any other function, an overloaded operator has a

return type and a parameter list.

Box operator+(const Box&amp;);

declares the addition operator that can be used to add two Box objects and returns final Box

object.

### Overloadable/Non-overloadableOperators:

Following is the list of operators which can be overloaded:

+ - \* / % ^

&amp; | ~ ! , =

Roll No. SC55 - Shreyas Chavhan

===================================================

+= -= /= %= ^= &amp;=

|= \*= &lt;&lt;= &gt;&gt;= [ ] ( )

-&gt; -&gt;\* new new [ ] delete delete [ ]

Operator that are not overloaded are follows

scope operator - ::

sizeof

member selector - .

member pointer selector - \*

ternary operator - ?:

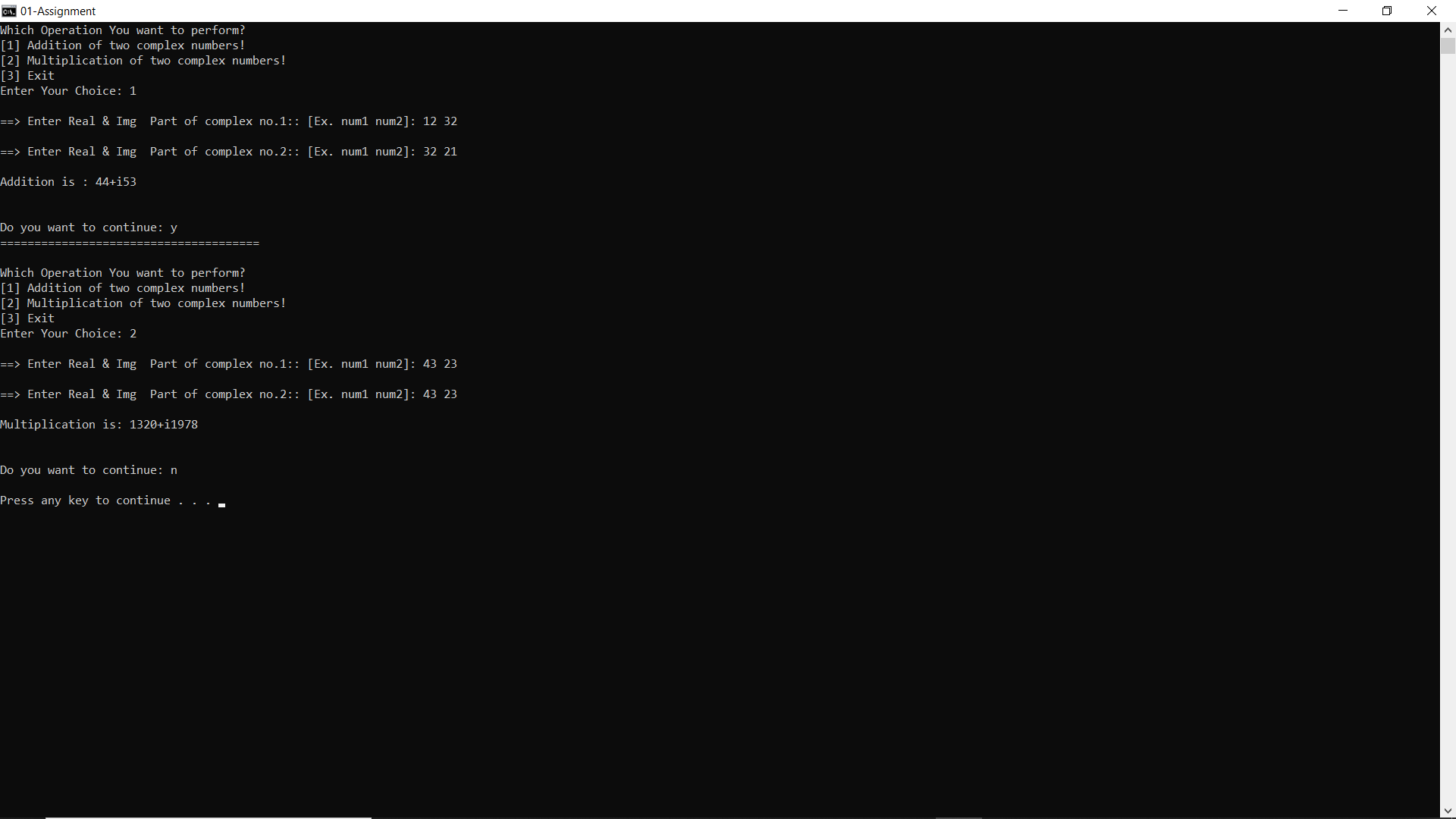
## Code -

|  |
| --- |
| /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Implement a class Complex which represents the Complex Number data type. Implement the following operations:  1. A 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 Number  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #include<iostream>  using namespace std;  class complex  {  float real, imag;  public:  complex(float x1, float x2) // parametrrised consructor  {  real = x1;  imag = x2;  }  complex() // default consructor #1  {  real = imag = 0;  }  friend istream &operator >> (istream &in,complex &t) //operator >> overloaded #4  {  in >> t.real >> t.imag;  return in;  }  complex operator + (complex t) //operator + overloaded #2  {  complex z;  z.real = this->real + t.real;  z.imag = this->imag + t.imag;  return z;  }  complex operator \* (complex t) //operator \* overloaded #3  {  complex z;  z.real = this->real \* t.real-this->imag \* t.imag;  z.imag = this->imag \* t.real+this->real \* t.imag;  return z;  }  friend ostream &operator << (ostream &op,complex &t) //operator << overloaded #4  {  op << t.real << "+i" << t.imag;  return op;  }  };  int main()  {  complex c1; // creates complex no 0+i0(default)  complex c2(3,5); // creates complex no 3+i5(parameterized)  complex c3,c4;  int c;  char ch;  do  {  cout << "======================================\n";  cout << endl;  cout << "Which Operation You want to perform?\n";  cout << "[1] Addition of two complex numbers!\n";  cout << "[2] Multiplication of two complex numbers!\n";  cout << "[3] Exit";  cout << endl;  cout << "Enter Your Choice: ";  cin >> c;  switch(c)  {  case 1:  cout << endl;  cout << "==> Enter Real & Img Part of complex no.1:: [Ex. num1 num2]: ";  cin >> c1;  cout << "\n==> Enter Real & Img Part of complex no.2:: [Ex. num1 num2]: ";  cin >> c2;  c3 = c1 + c2;  cout << endl;  cout << "Addition is : ";  cout << c3 << "\n";  cout << endl;  cout << endl;  break;  case 2:  cout << endl;  cout << "==> Enter Real & Img Part of complex no.1:: [Ex. num1 num2]: ";  cin >> c1;  cout << "\n==> Enter Real & Img Part of complex no.2:: [Ex. num1 num2]: ";  cin >> c2;  cout << endl;  cout << "Multiplication is: ";  c4 = c1 \* c2;  cout << c4 << "\n";  cout << endl;  cout << endl;  break;  case 3:  return 0;  }  cout << "Do you want to continue: ";  cin >> ch;  }while(ch == 'y' || ch == 'Y');  return 0;  } |

Roll No. SC55 - Shreyas Chavhan

===================================================

## Output -



===================================================

Roll No. SC55 - Shreyas Chavhan