```
1  //
2  // Shaun Chemplavil U08713628
3  // shaun.chemplavil@gmail.com
4  // C/C++ Programming III : Intermediate Programming with Objects
5  // 151116  Raymond L. Mitchell III
6  // Complex.h
7  // Win10
8  // Visual C++ 19.0
9  //
10 // File contains the SavingsAccount class definition
11 //
12
13 #ifndef SHAUNCHEMPLAVIL_COMPLEX_H
14 #define SHAUNCHEMPLAVIL_COMPLEX_H
15
16 #include<iostream>
17
18 using std::ostream;
19 using std::istream;
20
21 namespace ShaunChemplavil
22 {
23    class Complex
24    {
25       friend ostream &operator<<(ostream &out, const Complex &value);
26       friend istream &operator>>(istream &in, Complex &value);
27       friend Complex operator+(const Complex &op1, const Complex &op2);
28       friend Complex operator-(const Complex &op1, const Complex &op2);
29       friend bool operator==(const Complex &op1, const Complex &op2);
30       friend bool operator!=(const Complex &op1, const Complex &op2);
31
32    public:
33       // Constructors  (need to default member variables to 0.0)
34       Complex();
35       Complex(double real);
36       Complex(double real, double imaginary);
37
38    private:
39       double real;
40       double imaginary;
41    };
42 }
43
44 #endif
45
```

```cpp
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C/C++ Programming III : Intermediate Programming with Objects
// 151116  Raymond L. Mitchell III
// Complex.cpp
// Win10
// Visual C++ 19.0
//
// File contains the display member function for the SavingsAccount class
//

#include <iostream>
#include "Complex.h"

using std::ostream;
using std::istream;

namespace ShaunChemplavil
{
   // Default Constructor
   Complex::Complex()
      : real(0.0), imaginary(0.0) {}

   Complex::Complex(double real)
      : real(real), imaginary(0.0) {}

   Complex::Complex(double real, double imaginary)
      : real(real), imaginary(imaginary) {}

   ostream &operator<<(ostream &out, const Complex &value)
   {
      out << value.real;

      //Check if imaginary part is positive, and place '+'
      // (negative sign from imaginary will automatically be displayed)
      if (value.imaginary >= 0.0)
         out << "+";

      out << value.imaginary << "i";

      return out;
   }

   istream &operator>>(istream &in, Complex &value)
   {
      //a '+/-' separating the real from imaginary will put values in correct
      // member variables
      in >> value.real;
      in >> value.imaginary;

      // Ignore the character i
```

```
53          in.ignore();
54
55          return in;
56      }
57
58      Complex operator+(const Complex &op1, const Complex &op2)
59      {
60          return(Complex(op1.real + op2.real, op1.imaginary + op2.imaginary));
61      }
62
63      Complex operator-(const Complex &op1, const Complex &op2)
64      {
65          return(Complex(op1.real - op2.real, op1.imaginary - op2.imaginary));
66      }
67
68      bool operator==(const Complex &op1, const Complex &op2)
69      {
70          return((op1.real == op2.real) && (op1.imaginary == op2.imaginary));
71      }
72
73      bool operator!=(const Complex &op1, const Complex &op2)
74      {
75          return(!(op1 == op2));
76      }
77  }
78
```

```cpp
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C/C++ Programming III : Intermediate Programming with Objects
// 151116  Raymond L. Mitchell III
// hw4.cpp
// Win10
// Visual C++ 19.0
//
// Test Program for the Complex class
//

#include <iostream>
#include "Complex.h"
using std::cout;
using std::cin;

using ShaunChemplavil::Complex;

int main()
{
   // declare original date variables (arbitrary values)
   Complex input1, input2, temp, tempIn2(9.1, 8.55), tempIn1(1.2);
   cout << "Default input is: " << temp << "\n"
      << "Specified input (2 inputs) is: " << tempIn2 << "\n"
      << "Specified input (1 inputs) is: " << tempIn1 << "\n\n";

   cout << "Please enter 2 Complex numbers (space separated):\n";
   cin >> input1 >> input2;

   cout << "\nInput 1 is: " << input1 << "\n"
      << "Input 2 is: " << input2 << "\n\n"
      << "Input 1 + Input 2 + Specified (2 inputs) = " << (input1 + input2 +
       tempIn2) << "\n"
      << "Input 2 - Input 1 - Specified (2 inputs) = " << (input2 - input1 -
       tempIn2) << "\n\n"
      << "Input 1 == Input 1 :" << (input1 == input1) << "\n"
      << "Input 1 == Input 2 :" << (input1 == input2) << "\n"
      << "Input 1 != Input 2 :" << (input1 != input2) << "\n";

   return 0;
}
```

```
Microsoft Visual Studio Debug Console
Default input is: 0+0i
Specified input (2 inputs) is: 9.1+8.55i
Specified input (1 inputs) is: 1.2+0i

Please enter 2 Complex numbers (space separated):
2.5+5.6i 3.9+1.5i

Input 1 is: 2.5+5.6i
Input 2 is: 3.9+1.5i

Input 1 + Input 2 + Specified (2 inputs) = 15.5+15.65i
Input 2 - Input 1 - Specified (2 inputs) = -7.7-12.65i

Input 1 == Input 1 :1
Input 1 == Input 2 :0
Input 1 != Input 2 :1

C:\Users\schem\source\repos\schemp98\Cpp_Certification_Course\Debug\Exercise.exe
 (process 106548) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->
Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```