

## Object-Oriented Programming (3190)

### Homework 6

Spring 2023

1. [10pt] Design a Complex class representing complex numbers. A complex number in mathematics is defined as  $x + iy$  where  $x$  defines the real part of the number and  $y$  is the imaginary part. The letter  $i$  represents the square root of  $-1$  (which means  $i^2$  is  $-1$ ). Include operator overloading functions to calculate mathematical operations for the class. Note that the following relationships exist between two complex numbers:

$$\begin{aligned}(x_1 + i y_1) + (x_2 + i y_2) &= (x_1 + x_2) + i (y_1 + y_2) \\(x_1 + i y_1) - (x_2 + i y_2) &= (x_1 - x_2) + i (y_1 - y_2) \\(x_1 + i y_1) * (x_2 + i y_2) &= (x_1 x_2 - y_1 y_2) + i (x_1 y_2 + x_2 y_1) \\(x_1 + i y_1) / (x_2 + i y_2) &= ((x_1 x_2 + y_1 y_2) + i (-x_1 y_2 + x_2 y_1)) / (x_2^2 + y_2^2)\end{aligned}$$

Your solution must meet the following requirements:

- (a) Include operator functions to overload the operators  $+=$ ,  $-=$ ,  $*=$ ,  $/=$  for the class.
- (b) Include operator functions to overload the operators  $+$ ,  $-$ ,  $*$ ,  $/$  for the class.
- (c) Include operator function to overload the operator  $<<$  for printing complex numbers.
- (d) Make sure that your output matches the following output, and test your code with at least two more test cases on your own. It is preferred to test extreme cases to prove that your code is working correctly.

Possible Output:

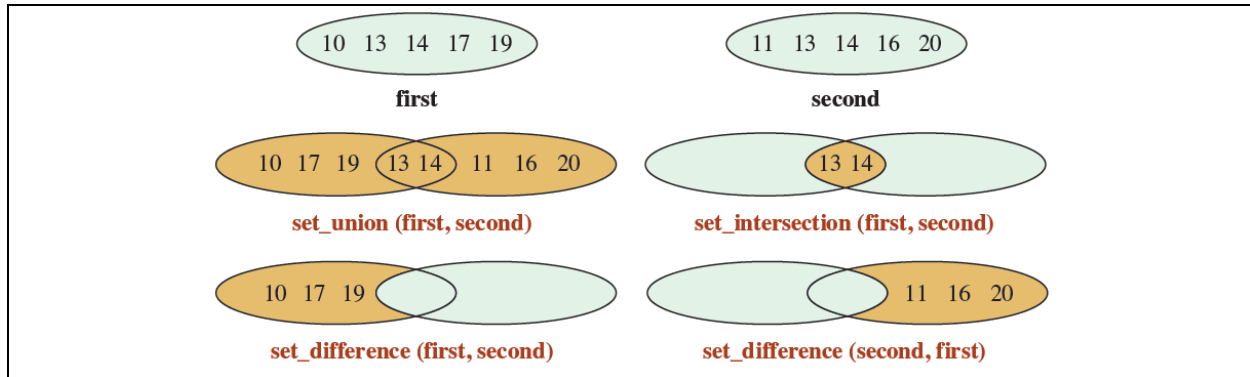
```
complex1: (+2.00) + i(+3.00)
complex2: (+2.00) + i(+3.00)
complex3: (+2.00) + i(+3.00)
complex4: (+2.00) + i(+3.00)
rvalue: (+1.00) + i(+2.00)
complex1 += rvalue: (+3.00) + i(+5.00)
complex2 -= rvalue: (+1.00) + i(+1.00)
complex3 *= rvalue: (-4.00) + i(+7.00)
complex4 /= rvalue: (+1.60) + i(-0.20)
resulting complex1 + rvalue: (+4.00) + i(+7.00)
resulting complex2 - rvalue: (+0.00) + i(-1.00)
resulting complex3 * rvalue: (-18.00) + i(-1.00)
resulting complex4 / rvalue: (+0.24) + i(-0.68)

#-- Custom Test Cases --
```

## Object-Oriented Programming (3190)

### Homework6

2. [10pt] Declare and define a class Set representing a set of integers. A set is a collection of data without repetition or ordering. The class should have **only private data members**: a pointer to a dynamically allocated array of integers and an integer that holds the size of the set. The following shows the operators to be defined for a set.



Your solution should have the following methods:

- (a) A constructor to create an empty set. A destructor. A copy constructor.
- (b) A function to add an element to the set (overload the += operator).
- (c) A function to remove an element from the set (overload the -= operator).
- (d) A binary friend function to get the union of two sets (overload the + operator).
- (e) A binary friend function to get the intersection of two sets (overload the \* operator).
- (f) A binary friend function to determine the difference of two sets (overload the - operator).
- (g) A function to print the contents of a set (overload the << operator).

Make sure that your output matches the following output, and test your code with at least two more test cases on your own. It is preferred to test extreme cases to prove that your code is working correctly.

Possible Output:

```
Set1:
Set1: 19 10 17
Set1: 19 10 17 20 21 13 14
Set1: 19 10 17 13 14
Set2:
Set2: 22 11 20
Set2: 22 11 20 16 13 14 23
Set2: 11 20 16 13 14
Union of set1 and set2: 19 10 17 13 14 11 20 16
Difference of set1 and set2: 19 10 17
Difference of set2 and set1: 11 20 16
Intersection of set1 and set2: 13 14

#-- Custom Test Cases --
```

End of Assignment.