Date: 29/Jan/2020

Lab Exercise
OOAD

Q1. Create a class called Complex for performing arithmetic with complex numbers. Complex numbers have the form
realPart + *imaginaryPart* * *i* where *i* is$\sqrt{-1}$

Write a program to test your class. Use floating-point variables to represent the private data of the class. Provide a constructor that enables an object of this class to be initialized when it is declared. Provide a no-argument constructor with default values in case no initializers are provided. Provide public methods that perform the following operations:
a) Add two Complex numbers: The real parts are added together and the imaginary parts are added together.
b) Subtract two Complex numbers: The real part of the right operand is subtracted from the real part of the left operand, and the imaginary part of the right operand is subtracted from the imaginary part of the left operand.
c) Print Complex numbers in the form (a, b), where a is the real part and b is the imaginary part.

Q2. Create class IntegerSet. Each IntegerSet object can hold integers in the range 0–100. The set is represented by an array of booleans. Array element a[i] is true if integer *i* is in the set. Array element a[j] is false if integer *j* is not in the set. The no-argument constructor initializes the Java array to the "empty set" (i.e., a set whose array representation contains all false values).

Provide the following methods: Method *union* creates a third set that is the set-theoretic union of two existing sets (i.e., an element of the third set's array is set to true if that element is true in either or both of the existing sets—otherwise, the element of the third set is set to false). Method *intersection* creates a third set which is the set-theoretic intersection of two existing sets (i.e., an element of the third set's array is set to false if that element is false in either or both of the existing sets—otherwise, the element of the third set is set to true). Method *insertElement* inserts a new integer *k* into a set (by setting a[k] to true). Method *deleteElement* deletes integer *m* (by setting a[m] to false). Method *toSetString* returns a string containing a set as a list of numbers separated by spaces. Include only those elements that are present in the set. Use --- to represent an empty set. Method *isEqualTo* determines whether two sets are equal.

Write a program to test class *IntegerSet*. Instantiate several *IntegerSet* objects. Test that all your methods work properly.