Total Score: 98/100
70/70

```
/**
 * @author Daniel Kim
 * The point of this was to work was to work with constructor objects setting up
real and imaginary numbers
 * There was 4 operations that we had to do that was the
sum,multiplication,subtraction and setting it negative number
 * Also did the get and set methods to retrieve real and imaginary numbers
 * Built constructor methods to give value or r to real numbers and i to imaginary
numbers
 */
package assg2_kimj19;


public class ComplexNum {

      private double real;
      private double imaginary;
      /**
       * Default constructor method
       */
      public ComplexNum() {
            real = 0;
            imaginary = 0;
      }
      /**
       * Constructor with a given r for radius
       * @param r is the given radius
       */
      public ComplexNum(double r) {
            real = r;
            imaginary = 0;
      }
      /**
       * Constructor with the given r and i (real and imaginary)
       * @param r given value of real
       * @param i given value of imaginary
       */
      public ComplexNum(double r, double i) {
            real = r;
            imaginary = i;
      }
      /**
       * It retrieves real
       * @return
       */
      public double getReal() {
            return real;
      }
      /**
       * it retrieves imaginary
       * @return
       */
      public double getImaginary() {
            return imaginary;
      }
      /**
       * Modifies real and sets it as newreal
       * @param newreal
       */
      public void setReal(double newreal) {
            real = newreal;
      }
      /**
       * Modifies imaginary and sets it as newimaginary
       * @param newimaginary
       */
      public void setImaginary(double newimaginary) {
            imaginary = newimaginary;
      }
      /**
       * The parameter add is addition. It adds up both of the real numbers and
imaginary numbers together
       * @param num (Complex Number)
       * @return
       * the return statement returns the addition of both real and imaginary
numbers
       */

      public ComplexNum add(ComplexNum num) {
            ComplexNum sum = new ComplexNum();
            sum.real = this.real + num.real;
            sum.imaginary = this.imaginary + num.imaginary;
            return sum;
      }
      /**
       * The parameter sub is subtraction. It subtracts both of the real numbers
and imaginary numbers together
       * @param num (complex number)
       * @return
       * the return statement returns the subtraction of both real and imaginary
numbers.
       */
      public ComplexNum sub (ComplexNum num) {
            ComplexNum sub = new ComplexNum();
            sub.real = this.real - num.real;
            sub.imaginary = this.imaginary - num.imaginary;
            return sub;
      }
      /**
       * The parameter mul is multiplication. It multiplies both of the real
numbers and imaginary numbers together
       * @param num (Complex number)
       * @return
       * the return statement returns the multiplication of both real and imaginary
numbers.
       */
      public ComplexNum mul (ComplexNum num) {
            ComplexNum mul = new ComplexNum();
            mul.real = (this.real * num.real) - (this.imaginary * num.imaginary);
            mul.imaginary = (this.real * num.imaginary) + (this.imaginary *
num.real);
            return mul;

      }
      /**
       * the parameter neg is setting the imaginary and real number to negative
values.
       * @param num (Complex number)
       * @return
       * Returns real and imaginary as negative integers
       */
      public ComplexNum neg (ComplexNum num) {
            ComplexNum neg = new ComplexNum();
            neg.real = -this.real;
            neg.imaginary =   -this.imaginary;
            return neg;
      }
      /**
       * toString method is to be able to print out real numbers and for imaginary
numbers it adds the i to the end
       * of the imaginary number.
       * @return real numbers and for imaginary number it adds the i at the
end(only imaginary) and is returning it
       * as a string
       */
      @Override
      public String toString() {
            if (imaginary == 0)
                  return real + "";
            if (real == 0)
                  return imaginary + "i";
            if (imaginary > 0 && real >0)
                  return real + "+" + imaginary+"i";
            return real + " - " + imaginary + "i";
      }
      /**
       * Compares the Complex num to another object
       * @param obj is the object for comparison
       * @return it returns true when the complex num is equal to a complex number
or otherwise it returns false
       *
       */
      @Override
      public boolean equals(Object obj) {
            if (obj==null) {
                  return false;
            }
            if(obj instanceof ComplexNum) {
                  ComplexNum temp = (ComplexNum)obj;
                  return (this.real == temp.real && this.imaginary ==
temp.imaginary);
            }
            else {
                  return false;
            }
      }

      }
```