ECET11 – Object Oriented Programming LAB activity 2

Objective: Classes Revisited

Source: Deitel & Deitel Edition 5 – Chapter 9 Exercises

1. (Complex Class) Create a class called Complex for performing arithmetic with complex numbers. Write a program to test your class. Complex numbers have the form:

   Real + ***j*** Imaginary

   where ***j*** $= \sqrt{-1}$

   Use double 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. The constructor should contain default values in case no initializers are provided. Provide public member functions that perform the following tasks:

   - Adding two Complex numbers: The real parts are added together and the imaginary parts are added together.

   - Subtracting 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.

   - Printing Complex numbers in the form (a, b)

     ✓ a is the real part
     ✓ b is the imaginary part.

2. (Rational Class) Create a class called Rational for performing arithmetic with fractions. Write a program to test your class.

   Use integer variables to represent the private data of the class - the numerator and the denominator. Provide a constructor that enables an object of this class to be initialized when it is declared.

The constructor should contain default values in case no initializers are provided and should store the fraction in reduced form. For example, the fraction would be stored in the object as 1 in the numerator and 2 in the denominator. Provide public member functions that perform each of the following tasks:

- Adding two Rational numbers. The result should be stored in reduced form.

- Subtracting two Rational numbers. The result should be stored in reduced form.

- Multiplying two Rational numbers. The result should be stored in reduced form.

- Dividing two Rational numbers. The result should be stored in reduced form.

- Printing Rational numbers in the form a/b, where a is the numerator and b is the denominator.

- Printing Rational numbers in floating-point format.

3. (Rectangle Class) Create a class Rectangle that stores the Cartesian coordinates of the four corners of a given rectangle. The constructor calls a set function that accepts four sets of coordinates and verifies that each of these is in the first quadrant with no single x- or y-coordinate larger than 20.0. The set function also verifies that the supplied coordinates do, in fact, specify a rectangle. Provide member functions that calculate the length, width, perimeter and area. The length is the larger of the two dimensions. Include a predicate function square that determines whether the rectangle is a square.

- Provide a draw function that displays the rectangle. Include a setFillCharacter function to specify the character out of which the body of the rectangle will be drawn.

- Provide a function to scale the size of the rectangle

- Provide a function to rotate the rectangle

4. (TicTacToe Class) Create a class **TicTacToe** that will enable you to write a complete program to play the game of tic-tac-toe. The class contains as **private** data a 3-by-3 array of chars. A constructor initializes the empty board to all blanks. Allow two human players (x and o). Wherever the x player moves, place an x in the specified square; place an o wherever the o player moves. Each move must be to an empty square. After each move, determine if the game has been won or if the game is a draw.