Homework 5:

Submit your work (the .java source code files ONLY, not the compiled .class files!) through the "Homework5" link on Brightspace. You may submit an unlimited number of times; we will only grade the last/latest submission attempt, but be sure to attach all of your files to each submission attempt. Be sure to include your name and Stony Brook ID number in a comment at the beginning of each file that you submit.

Due: Sunday, November 17, 2024 11:59 PM New York Time.

Points: This assignment is worth 50 points (10 + 10 + 10 + 10 + 10).

Submission Instructions: Name your java classes for this assignment as:

Problem1: MatrixMultiply.java
Problem2: MatrixSort.java
Problem3: Complex.java
Problem4: Rational.java
Problem5: Circle2D.java

1. Write a method to multiply two matrices. The header of the method is:
   $public\ static\ double[][]\ multiplyMatrix(double\ [][]\ a, double[][]\ b)$.
   To multiply matrix a by matrix b, the number of columns in a must be the same as the number of rows in b, and the two matrices must have elements of the same or compatible types. Let c be the result of the multiplication. Assume the column size of matrix a is n. Each element $c_{ij} = a_{i1} \times b_{1j} + a_{i2} \times b_{2j} + \cdots + a_{in} \times b_{nj}$.
   For example, the two 3×3 matrices a and b, c is:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix}$$

   Where $c_{ij} = a_{i1} \times b_{1j} + a_{i2} \times b_{2j} + a_{i3} \times b_{3j}$.
   Write a test program that prompts the user to enter two 3×3 matrices and displays their product.

   Here is a sample run:

```
Enter matrix1: 1 2 3 4 5 6 7 8 9
Enter matrix2: 0 2 4 1 4.5 2.2 1.1 4.3 5.2
Multiplication of the matrices is:  5.3    23.9    24
                                    11.6   56.3    58.2
                                    17.9   88.7    92.4
```

2. Implement the following method to sort (in ascending order) the rows in a two-dimensional array. A new array is returned and the original array is intact.

```
public static double[][] sortRows(double[][] m)
```

Write a test program that prompts the user to enter a 3×3 matrix of double values and displays a new row-sorted matrix. Here is a sample run: [10]

Enter a 3-by-3 matrix row by row:
0.15 0.875 0.375
0.55 0.005 0.225
0.30 0.12 0.4

The row-sorted array is:
0.15 0.375 0.875
0.005 0.225 0.55
0.12 0.30 0.4

[Note: Please implement your own sorting algorithm. You can implement Bubble sort.]

3. 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 double variable to represent the private data of the class. Provide a constructor that enables an object of this class to be initialized when it's 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 the real part and b is the imaginary part.

[10]

Here is a sample run:

Enter real part of the first complex number: 3
Enter imaginary part of the first complex number: 6
Enter real part of the second complex number: 2
Enter imaginary part of the second complex number: 7

First complex number is: (3.0, 6.0)
Second complex number is: (2.0, 7.0)
Addition of the complex numbers is: (5.0, 13.0)
Subtraction of the complex numbers is: (1.0, -1.0)

4. Create a class called Rational for performing arithmetic with fractions. Write a program to test your class. Use integer variables to represent the private instance variables of the class - the numerator and the denominator. Provide a constructor that enables an object of this class to be initialized when it's declared. The constructor should store the fraction in reduced form. The fraction

   $2/4$

   is equivalent to $1/2$ and would be stored in the object as 1 in the numerator and 2 in the denominator. Provide a no-argument constructor with default values in case no initializers are provided. Provide public methods that perform each of the following operations:

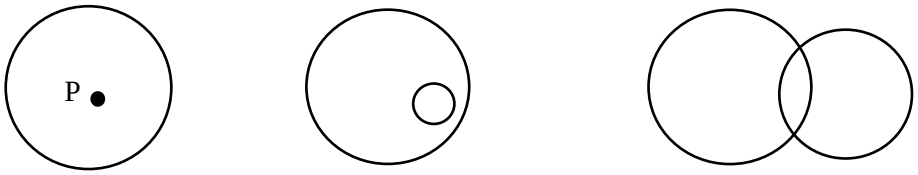
   a) Add two Rational numbers: The result of the addition should be stored in reduced form.
   b) Subtract two Rational numbers: The result of subtraction should be stored in reduced form.
   c) Multiply two Rational numbers: The result of the multiplication should be stored in reduced form.
   d) Divide two Rational numbers: The result of the division should be stored in reduced form.
   e) Return a String representation of Rational number in the form a/b, where a is the numerator and b is the denominator.

   Here is a sample run:
   Enter numerator for the first rational number: 4
   Enter a non-zero denominator for the first rational number: 6
   Enter numerator for the second rational number: 1
   Enter a non-zero denominator for the second rational number: 4
   First rational number is: 2/3
   Second rational number is: 1/4
   Addition of the rational numbers is: 11/12
   Subtraction of the rational numbers is: 5/12
   Multiplication of the rational numbers is: 1/6
   Division of the rational numbers is: 8/3 [10]

5. Define the Circle2D class that contains:
   a. Two double data fields named x and y that specify the center of the circle with getter methods.
   b. A data field radius with a getter method.
   c. A no-arg constructor that creates a default circle with (0, 0) for (x, y) and 1 for radius.
   d. A constructor that creates a circle with the specified x, y, and radius.

e. A method getArea() that returns the area of the circle.
f. A method getPerimeter() that returns the perimeter of the circle.
g. A method contains(double x, double y) that returns true if the specified point (x, y) is inside the circle (see the figure below).
h. A method contains (Circle2D circle) that returns true if the specified circle is inside this circle (see the figure below).
i. A method overlaps(Circle2D circle) that returns true if the specified circle overlaps with this circle (see the figure below).

Write a test program that creates a Circle2D object c1 with its x, y coordinates and the radius as 2, 2, and 5.5, respectively. Display the area and the perimeter of c1. Display the result of c1.contains(3, 3), c1.contains (new Circle2D(4, 5, 10.5)), and c1.overlaps(new Circle2D(3, 5, 2.3)).



(a) (b) (c)

Figure: (a): A point inside the circle. (b): A circle inside another circle. (c): A circle overlaps another circle. [10]