Data Structures and Objects
CSIS 3700

Spring Semester 2018 — CRN 21212

Project 1 — Fractions
Due date: Friday, February 9, 2018

Goal

Develop and implement a **`Fraction`** class and use it to calculate the area of a polygon.

Details

▹*The `Fraction` class*

For this project, you should use the **`Fraction`** class you created in lab.

▹*Source code layout*

In this project you should use a conventional text editor and the **`g++`** and **`make`** tools. Since I want you to get some additional experience with these tools, you should not use an IDE for this project.

The project is also an exercise in *separate compilation*, where the source code is divided into multiple parts. The project consists of four files:

- A **`fraction.h`** header file that contains the class definition;
- A **`fraction.cc`** file containing the class methods (functions);
- A file containing a **`main`** function that performs the intersection calculations
- A **`Makefile`** that directs the process of creating the executable file

▹*Calculating area*

To calculate the area, you must first be able to calculate the *cross product* of two points. Given $\mathbf{p_1} = (x_1, y_1)$ and $\mathbf{p_2} = (x_2, y_2)$, the cross product is $\mathbf{p_1} \times \mathbf{p_2} = x_1 y_2 - x_2 y_1$.

Now, suppose you have a polygon whose vertices are $\langle \mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_n} \rangle$, where the vertices are listed consecutively as you travel counterclockwise around the polygon. Then, the area of the polygon is given by

$$A = \frac{1}{2} \sum_{i=1}^{n} (\mathbf{p_i} \times \mathbf{p_{i+1}})$$

where $\mathbf{p_{n+1}} = \mathbf{p_1}$.

An easy way to perform the calculation is to keep track of three points: the first point in the list, the current / most recently input point, and the previous point.

Read the first point and also make that the current point. Then, perform a **`do`** loop. Inside, do the following:

- Copy the current point to the previous point
- Read the next (new "current") point
- Calculate the cross product, where $\mathbf{p_i}$ is the previous point and $\mathbf{p_{i+1}}$ is the current point. Add that to a running total

Stop the loop when the current point is the same as the first point. Then, divide the total by 2 and that will be the area.

Extra Credit

For 10% extra credit, write a **`Point`** class that implements the following:

- Input and output of a point in the form (x, y) where x and y are two **`Fraction`** objects
- Add and subtract two points using **`operator+`** and **`operator-`**
- Multiply two points, computing the cross product
- Multiply a point by a **`Fraction`**, which multiplies both of the point's coordinates by the given fraction.

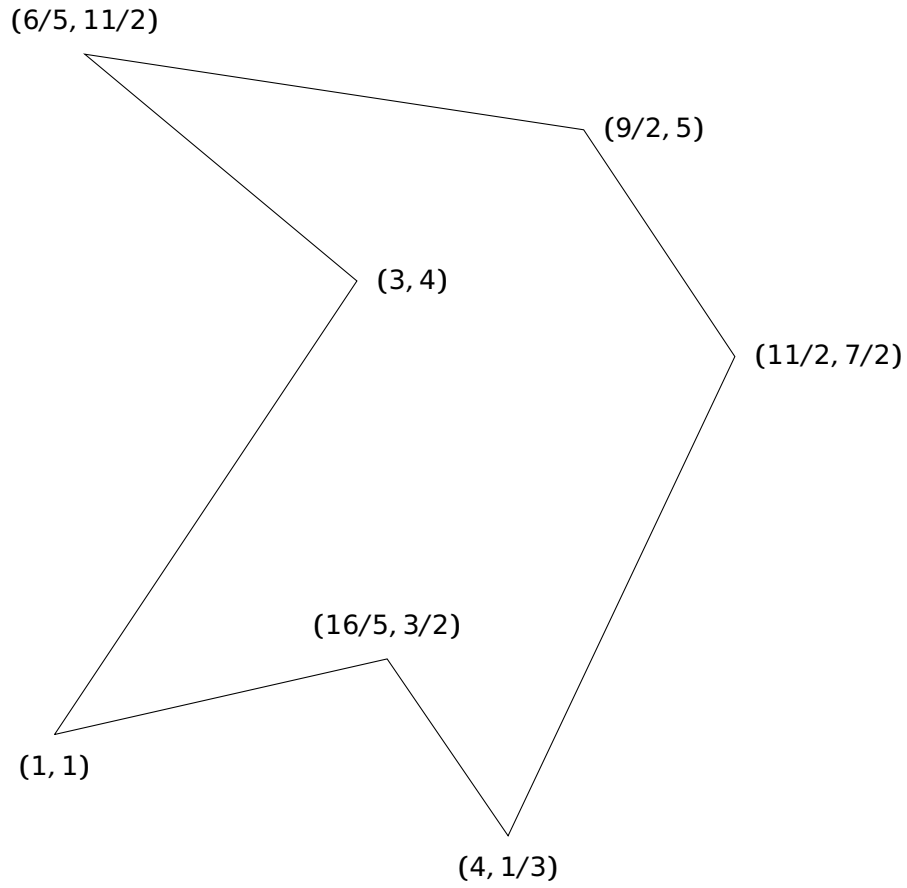
Note that both forms of multiply should use **`operator*`**; for cross product the parameter should be a **`Point`** and for scaling it should be a **`Fraction`**.

Note that you must use your **`Point`** class in the project's solution in order to receive the extra credit.

What to turn in

Turn in your source code and **`Makefile`**.

Example



Example Input

▹*Without a Point class*

```
1/1 1/1
16/5 3/2
4/1 1/3
11/2 7/2
9/2 5/1
6/5 11/2
3/1 4/1
1/1 1/1
```

▹*With a Point class*

```
(1/1,1/1)
(16/5,3/2)
(4/1,1/3)
(11/2,7/2)
(9/2,5/1)
(6/5,11/2)
(3/1,4/1)
(1/1,1/1)
```

Example Output

```
Area of polygon: 35 / 3
```