**CSE 174 – Sections E-H – Fall 2016**

**PROGRAM #4: 30 points – Due Sunday, September 25, by 11:59 p.m.**

**Outcomes:**

* Write programs that obtain user input
* Write programs that compute mathematical results
* Write programs that work with Strings
* Write programs that display numbers formatted according to a given specification
* Format and comment source code that adheres to a given set of formatting guidelines

**Scoring:**

At a bare minimum, the program you submit must have the assigned source code, and your source code must compile and run without crashing.

* If you do not submit a zip file containing your source code (.java files), your score will be zero.
* If you submit source code that does not compile, your score will be zero.
* If you submit source code that roughly resembles the requirements and it compiles, but it crashes under normal operating conditions (nice input from the user), your score will be reduced by 75%.
* Deductions will be made for not meeting the usual requirements:
* Source code that is not formatted according to guidelines (see program2 for more details)
* File and class names that do not meet specifications

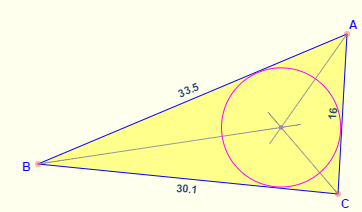
|  |  |  |
| --- | --- | --- |
|  | **Full credit** | **No credit or Partial credit** |
| **Get console input**  **(8 points)** | Using a Scanner, the program obtains user input, formatted as specified | Program does not use keyboard input, or uses it in a manner inconsistent with specifications |
| **Compute triangle measures**  **(12 points)** | Program correctly computes all mathematical measures (side lengths, perimeter, area, centroid, incircle radius, and incircle area) | There are mathematical errors in computation |
| **Format console output**  **(10 points)** | Screen output is formatted as specified, including converting vertex letters to uppercase | Screen output does not match specifications |

**Preliminaries:**

Before beginning to code, use the following website, <http://tube.geogebra.org/m/1601355>, to experiment with the triangle. This will be useful as you code, because it will let you create triangles to test with your Java program. Note that:

* The sketch shows the area and perimeter of the triangle.
* The left side of the screen shows the coordinates of the three vertices of the triangle, as well as the length of each side.
* The centroid of the triangle has been marked, and its coordinates can be found on the left side of the screen. The centroid is the "balance point" of a shape. If you were to cut out the shape and try to balance it on your finger, the centroid is where your finger would need to be.
* You can drag the triangle's vertices and all the measurements are automatically updated.
* You can also manually type in the coordinates of each vertex using the left side of the screen. This also automatically updates all measurements.

Also, do a little research on incircles. The incircle of a triangle is the largest **circle** that can fit inside it. An example of a triangle and its incircle are shown here:



Also, you need to be able to extract letters from a String and to convert a String to uppercase. The charAt(), substring(), and toUpperCase() methods are useful. You may choose to use one or more of these in the assignment.

**Requirements: You will write one Java class, TriangleMath**

Your program should:

* Prompt the user to enter the three-letter name of a triangle. Each letter names a vertex of the triangle.
* Prompt the user to enter the coordinates for the three vertices of that triangle (given in order corresponding to the letters the user entered).
* Display various calculations for that triangle, including:
  + The name and length of each side
  + The perimeter and area
  + The coordinates of the triangle's centroid (the centroid is the triangle's "center of mass", or balance point)
  + The length of the radius of the triangle's incircle
  + The area of the triangle's incircle

Below is a sample run of the program. Your program should be formatted as close to that sample as possible, including spacing, rounding (all values rounded to three places after the decimal point), and so on. The more closely you match, the better. Note that the user input is shown in bold and red. You are NOT supposed to print those. Those are typed by the user.

**Submission requirement (2 point deduction for incorrect submission):**

* Create a folder named program4.
* Inside that folder place exactly one file: TriangleMath.java
* Zip the program4 folder (not just the file in the folder, but the entire folder).
* Name the zip file program4.zip.

**Sample run. Your program should match this format as closely as possible**

|  |
| --- |
| Enter a three letter name for your triangle: **jqc**  Coordinates of vertex J: **.1 .1**  Coordinates of vertex Q: **2.2 .1**  Coordinates of vertex C: **1.15 .7**    --- Side lengths ---  JQ: 2.100  QC: 1.209  CJ: 1.209  --- Other measures ---  Perimeter = 4.519  Area = 0.630  Centroid = (1.150, 0.300)  Incircle radius = 0.279  Incircle area = 0.244 |