

Due: Monday, February 1, 2016 by 11:59 PM

Deliverables

Your project files should be submitted to Web-CAT by the due date and time specified above (see the Lab Guidelines for information on submitting project files). In order to avoid a late penalty for the project, you must submit your completed code files to Web-CAT no later than 11:59 PM on the due date for the completed code. You may submit your project up to 24 hours after the due date, but there is a late penalty of 15 points. No projects will be accepted after the one day late period. If you are unable to submit via Web-CAT, you should e-mail your project Java files in a zip file to your lab instructor before the deadline.

Files to submit to Web-CAT:

- SquareInches.java
- SlopeIntercept.java

Specifications

Overview: You will write two programs this week. The first will calculate the integral number of units (acres, square yards, and square feet) in a specified number of square inches, and the second will computer the Y and X intercepts for line given the X and Y coordinates for two points on the line.

- **SquareInches.java**

Requirements: An surveyor would like a program that allows the user to enter a value in square inches and then displays the combination of acres, square yards, square feet, and square inches so that each value is maximized in order by acres, square yards, square feet, and finally square inches. The input value should not exceed 1,000,000,000 square inches.

Design: The surveyor would like the output to look as shown below when 1234567890 is entered as the input for one run and 123456789 is entered for another run.

Line number	Program output
1	Enter the area in square inches: 1234567890
2	Limit of 1,000,000,000 square inches exceeded!

Line number	Program output
1	Enter the area in square inches: 123456789
2	Number of Units:
3	Acres: 19
4	Square Yards: 3299
5	Square Feet: 7
6	Square Inches: 117

Your program must follow the above format with respect to the output. Note that lines 3 through 6 for the input value of 123456789 begin with tab (i.e., your output should use the **escape sequence for a tab**).

Code: In order to receive full credit for this assignment, you must calculate the number of each unit (acres, square yards, square feet, square inches) and store the value in a variable of type *int*. It is recommended as a practice that you do not modify input values once they are stored. Note that an acre contains 6,272,640 square inches, a square yard contains 1,296 square inches, and a square foot contains 144 square inches. Your expressions should contain only *int* variables or *int* literals. Commas are not allowed in Java numeric literals (1,296 may be entered as 1296 or 1_296) (6,272,640 may be entered as 6272640 or 6_272_640).

Test: You will be responsible for testing your program, and it is important to not rely only on the example above. The amount entered can be any valid *int* value.

- **SlopeIntercept.java**

Requirements: A program is needed that takes the x and y coordinates of two points as doubles and calculates the slope (if defined) of the line defined by the two points. The slope is only defined if the two x coordinates are not equal. If the slope is defined, the program calculates and prints the Y intercept and then further, if the slope is not 0, it calculates and prints the X intercept.

Design: The equations for finding the slope (where $x_1 \neq x_2$) and the Y and X intercepts are as follows:

$$\text{slope} = \frac{y_2 - y_1}{x_2 - x_1} \quad \text{where } x_1 \neq x_2$$

$$y\text{intercept} = y - \text{slope} * x$$

$$x\text{intercept} = -1 * \frac{y\text{intercept}}{\text{slope}} \quad \text{where slope} \neq 0; \text{ i.e., } y_1 \neq y_2$$

Three examples of program output for the indicated input values are show below (replace everything in *italics* with your own words):

Example #1

Line number	Program output
1	<i>Enter the X and Y coordinates of starting point:</i>
2	x1 = 3
3	y1 = 1
4	<i>Enter the X and Y coordinates of ending point:</i>
5	x2 = 4
6	y2 = 2
7	Slope: 1.0
8	Y intercept: -2.0
9	X intercept: 2.0

Note that lines 2, 3, 5, and 6 begin with tab. Below is an example where slope is undefined.

Example #2

Line number	Program output
1	<i>Enter the X and Y coordinates of starting point:</i>
2	x1 = 2
3	y1 = 4
4	<i>Enter the X and Y coordinates of ending point:</i>
5	x2 = 3
6	y2 = 4
7	Slope: 0.0
8	Y intercept: 4.0
9	X intercept: "undefined"

Example #3

Line number	Program output
1	<i>Enter the X and Y coordinates of starting point:</i>
2	x1 = 2
3	y1 = 5
4	<i>Enter the X and Y coordinates of ending point:</i>
5	x2 = 2
6	y2 = 9
7	Slope: "undefined"

Code: Your numeric variables should be of type double. Use an if-else statement to determine if x_1 is equal to x_2 . If true, print a line indicating that the slope is undefined as shown in Example #3 above; if false, print value of the slope followed by the Y and X intercepts as shown in Example #1 above. Note that in Example #2, the X intercept was undefined since y_1 is equal to y_2 (e.g. requires an if-else statement to check for slope equal to zero).

Test: You are responsible for testing your program, and it is important to not rely only on the examples above. Remember that the input coordinates, slope, and the Y and X intercepts are doubles, so be sure to test both positive and negative values (with and without a decimal point) for the X and Y coordinates. You should use a calculator or jGRASP interactions to check your answers.

Grading

Web-CAT Submission: You must submit both “completed” programs to Web-CAT at the same time. Prior to submitting, be sure that your programs are working correctly and that they have passed Checkstyle. **If you do not submit both programs at once, Web-CAT will not be able to compile and run its test files with your programs which means the submission will receive zero points for correctness.** I recommend that you create a jGRASP project and add the two files. Then you will be able to submit the project to Web-CAT. Activity 1 (pages 5 and 6) describes how to create a jGRASP project containing both of your files.