

Deliverables

Your project files should be submitted to Web-CAT by the due date and time specified. In order to avoid a late penalty for the project, you must submit your completed code files to Web-CAT by 11:59 p.m. on the due date. 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 TA before the deadline.

Files to submit to Web-CAT:

- AreaOfPyramid.java
- LiquidMeasures.java

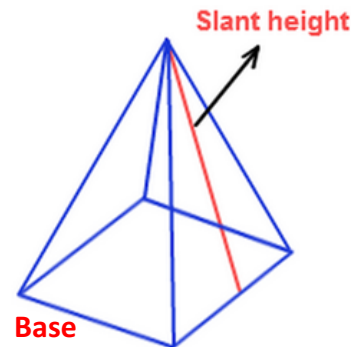
Specifications

Overview: You will write two programs this week. One will determine the area of a pyramid given the *base* and *slant height* of the pyramid. The other will calculate the numbers of barrels, gallons, quarts, and ounces in the number of liquid ounces entered by the user. The valued entered by the user must not exceed one billion.

- **AreaOfPyramid.java**

Requirements: A program is needed that reads in the *base* and *slant height* of a square-based pyramid, and then calculates the area of the pyramid using the following formula.

$$area = base \times base + 4 \left(\frac{base \times slant\ height}{2} \right)$$



Design: The main method of the program should allow the user to enter the values representing the pyramid as shown below, and then read in the values and store them in appropriately named variables of type double. After the values are read, the formula above is used to compute the area, after which the results are printed as shown below.

The following examples show the program input/output (I/O). Values entered by the user are indicated by red font.

Example 1

Line #	Program output
1	Enter values for base and slant height of a pyramid:
2	base = 10
3	slant height = 15
4	
5	A pyramid with base = 10.0 and slant height = 15.0
6	has an area of 400.0 square units.
7	

Note that lines 2 and 3 begin with tab (i.e., output should use the **escape sequence for a tab**. Line 4 is a blank line – consider printing a newline escape sequence (`\n`).

Example 2

Line #	Program output
1	Enter values for base and slant height of a pyramid:
2	base = 12.75
3	slant height = 12.75
4	
5	A pyramid with base = 12.75 and slant height = 12.75
6	has an area of 487.6875 square units.
7	

Code: Create a Scanner object on `System.in` to read in the values for the pyramid. The values of type double should be read in using the `nextDouble` method in the Scanner class.

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

- **LiquidMeasures.java**

Requirements: An oil company would like a program that allows the user to enter an amount of liquid in ounces, which must not exceed 1 billion, and then displays the number of barrels, gallons, quarts, and ounces. The number of each of these should be maximized from largest to smallest as indicated in Examples 2 and 3 below. Your program should use the following conversion values in the computation:

1 barrel = 42 gallons, 1 gallon = 128 ounces; 1 quart = 32 ounces.

Design: The oil company would like for the program's I/O to be as shown in the three example runs below where (1) 1234567890 is entered, (2) 12345 is entered, and (3) 12345678 is entered.

Example 1

Line #	Program output
1	Enter amount of liquid in ounces: 1234567890
2	Amount must not exceed 1,000,000,000.

Example 2

Line #	Program output
1	Enter amount of liquid in ounces: 12345
2	Measures by volume:
3	Barrels: 2
4	Gallons: 12
5	Quarts: 1
6	Ounces: 25
7	12345 oz = (2 bl * 5376 oz) + (12 gal * 128 oz) + (1 qt * 32 oz) + (25 oz)

Example 3

Line#	Program output
1	Enter amount of liquid in ounces: 12345678
2	Measures by volume:
3	Barrels: 2296
4	Gallons: 18
5	Quarts: 2
6	Ounces: 14
7	12345678 oz = (2296 bl * 5376 oz) + (18 gal * 128 oz) + (2 qt * 32 oz) + (14 oz)

Your program must follow the above format with respect to the output. Note that lines 3 through 6 for Examples 2 and 3 begin with tab (i.e., output should use the **escape sequence for a tab**).

Code: The input and output variables should be declared as type *int*. A simple if-else statement can be used to check that the amount does not exceed one billion, where the true block prints the error message and the false block prints the normal output (or vice-versa). Also, the return statement (`return;`) can be used in an if statement to return from main to immediately end the program after the error message. In order to receive full credit for this assignment, you must calculate the number of barrels, gallons, quarts, and ounces and store them in separate variables. It is recommended as a practice that you do not modify input values once they are stored.

Test: You will be responsible for testing your program, and it is important to not rely only on the example above. For example, test with the following amounts: 16, 32, 128, and 5376. The last amount is the number ounces in a barrel ($42 * 128$).

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 7 and 8) describes how to create a jGRASP project containing both of your files.