

Introduction to Computers and Engineering Problem Solving

Spring 2005

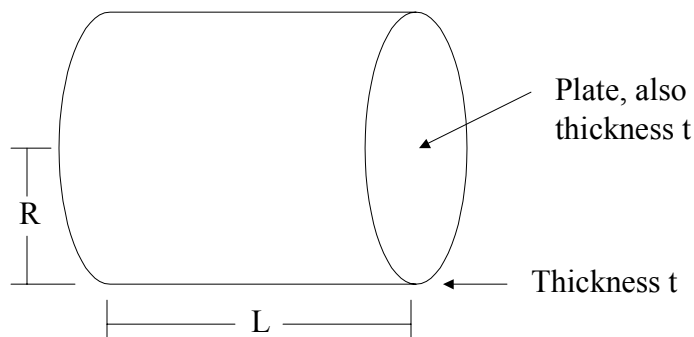
Problem Set 3: Tank design

Due: 12 noon, Session 11

1. Problem statement

You are designing a series of cylindrical tanks to hold hazardous waste. You must estimate the costs and volumes of the tanks, and keep other associated data such as their name or ID. You will be writing a Tank class in Java to implement each tank as an object with member data that describe its state or attributes, and member methods that describe its behaviors.

The tanks consist of a cylinder with plates welded on each end (Figure 1).



Tank dimensions

$$\text{The tank volume is } V = \pi R^2 L \quad (1)$$

$$\text{The tank cost is } C = c_1 m + c_2 w \quad (2)$$

where m = mass of steel used in tank (kg)
 w = length of welds needed to fabricate tank (m)
 c_1 = cost of steel, per kg
 c_2 = cost of welding, per m

The amount of material in the tank, m , is given by:

$$m = \rho(V_1 + V_2) \quad (3)$$

$$V_1 = L\pi[(R+t)^2 - R^2] \quad (4)$$

$$V_2 = 2\pi(R+t)^2 t \quad (5)$$

where V_1 = volume of material in the cylinder walls (m^3)
 V_2 = volume of material in the end plates (m^3)
 ρ = density of steel (kg/m^3)
 t = thickness of cylinder walls (m)

The length of the weld to attach the end plates to the cylinders is the sum of the cylinders' inside and outside diameters. It takes two welds per end plate to attach it to the cylinder.

$$w = 2 [2\pi(R+t) + 2\pi R] = 8\pi(R + t/2) \quad (6)$$

Parameter values:

Tank thickness	t	0.04m
Density of steel	ρ	8000 kg/m^3
Cost of steel	c_1	\$6/kg
Cost of welding	c_2	\$30/m
Minimum radius	R_{min}	0.5 m
Maximum radius	R_{max}	2m
Radius increment	R_{incr}	0.1m

The variables are R and L , the radius and length of the tank. The volume V will be specified by the user for each tank. Your program, among other things, will search for the lowest cost design for each tank using a simple strategy:

- a. Start searching at $R = R_{min} = 0.5m$
- b. Compute $L = V/\pi R^2$
- c. Compute the tank cost C
- d. Increment R by 0.1m, the minimum increment and loop until $R = R_{max}$, recording R , L and C at each value of R . Choose the minimum cost solution. Be careful that your loop terminates correctly (rounding error).

2. Problem objective and approach

Create files to implement two Java classes, `Tank.java` and `TankTest.java`

1. `Tank.java` : Your program must do the following:
 - Must have at least one constructor.

- Must store the following member data: (Use the correct access specifier (public or private) for each data member).
 - volume,
 - tank name
 - tank cost,
 - radius,
 - length,
 - mass of steel
 - length of welds
- Must store the following as static final variables (we will cover the static keyword shortly).
 - thickness,
 - steel density,
 - Rmin, Rmax, Rincr
- DO NOT store the cost parameters c1 or c2.
- Must have the following instance methods (each method must have appropriate arguments, return values and correct access specifier (public or private))
 - get() methods for volume, tank name, tank cost, radius, length, steel mass and weld length.
 - set() method for tank name. Do not have set methods for the other variables (why not?).
 - computeCost() : This method takes 2 arguments, c1 and c2 and finds the minimum cost tank for the volume (*based on the strategy given in the problem statement on page 2*). In so doing, it also finds and sets the following: tank cost, radius, length of tank, mass of tank and weld length.
- Must initialize all the data members to some unusual number (Double.MAX_VALUE, for example) to ensure that bogus values are not returned if a user doesn't call the computeCost() method before getting tank parameters.

2. TankTest.java: This class has only a main() method. This method will:

- Set the values of c1 and c2 (you don't need to accept them as inputs)
- Accept inputs for the number of tanks (a maximum of 3 tanks)
- For each tank, accept input for its:
 - Volume
 - Tank name (a string)
- Create the number of Tank objects requested.
- Compute and output each tank's name, volume, radius, length, cost, steel mass and weld length.
- Compute and output the sum of all tanks volume, costs, steel mass and weld length.

You don't need to check any inputs for validity.

Turn In

1. Place a comment with your full name, MIT server username, section, TA name and assignment number at the beginning of all `.java` files in your solution.
2. Place all of the files in your solution in a single zip file.
 - a. Do not turn in electronic or printed copies of compiled byte code (`.class` files) or backup source code (`.java~` files)
 - b. Do not turn in printed copies of your solution.
3. Submit this single zip file.
4. Your solution is due at noon. Your uploaded files should have a timestamp of no later than noon on the due date.

Penalties

- 30% off if you turn in your problem set after Friday noon but before noon on the following Monday. You have one no-penalty late submission per term for a turn-in after Friday noon and before Monday noon.
- No credit if you turn in your problem set after noon on the following Monday.