Homework 2 - Read from a File and Add Surface Area

In this assignment, you modifying Homework 1 to read the input data from a file and add surface area calculations outupt to Homework 1.

Part I: Calculating Areas

1. Follow steps similar to Program 1.

2. Include the following like in the exercise:

   ```
   #define _USE_MATH_DEFINES // for C++
   #include <cmath>
   #include <iostream>
   #include <iomanip>
   #include <fstream>
   ```

   And use the constant **M_PI** for **π** in your equations.

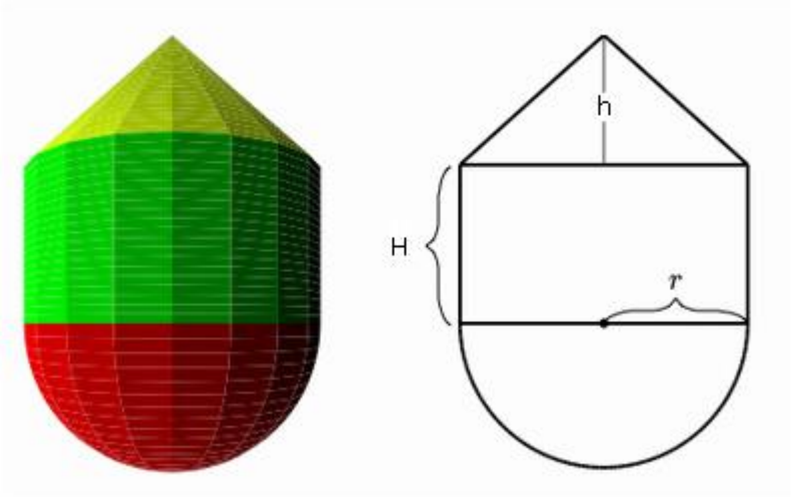
3. You do not prompt and read all of the data for this assignment. You only **prompt is for the name of the input file**. Then, the remainder of the input is read from that input file, e.g.

   ```
   Name of input file: infile.in
   ```

   This file contains only:

   ```
   3.3     4.7    11.9
   ```

4. You can leave the prompts in from your previous homework although they are ignored when the program in reading input data from a file. Use type **double** for all of the variables. You are calculating the volume of a tank shaped like a cone on top of a cylinder with ½ of a sphere on the bottom.



Use the following formulas in your program, with new surface area formulas:

Volume of a cone is $\frac{1}{3} \times \pi \times r^2 \times h$

SurfaceArea of a cone is π r $\sqrt{(r^2+h^2)}$ (only the top, not the base)

Volume of a cylinder is $\pi \times r^2 \times$ H

SurfaceArea of a cylinder is 2 π r H (only the "curve sides", not the top & bottom circles)

Volume of a **whole** sphere is $\frac{4}{3} \times \pi \times r^3$

SurfaceArea of a **whole** sphere is 4 π r^2

5. You need to read input for the radius (**r** in the diagram) of the cone, cylinder, and ½ sphere; for the height of the cone (**h** in the diagram); and for the height of the cylinder (**H** in the diagram), and . **r** is the same for all three! [You should use variable names better than r, h, and H!!! like coneHeight] The order of the data in the file is radius, cone height, and cylinder height.
6. Use "cin-type" statements, i.e. use the extraction operator. You may not use getline(), getc(), getchar(), etc.
7. Do not forget to declare inside main() all the variables to store values for inputs as well as for the calculated volume values AND surface area values for cone, cylinder, ½ sphere, and combined/total.
8. Please use incremental development by adding a few lines of code at a time and running it to make sure it is OK. Do not write the whole program, then try to compile it. Write only a few lines at a time, compile and fix that part, then add more.

`toddin.txt` contains: `3.3      4.7  11.9`

Your output should look like this: (Please copy and paste the text from these prompts into your code! You are graded on this on Web-CAT.)

```
Name of input file: toddin.txt

The volume of the cone portion is 53.599.
The volume of the cylinder portion is 407.122.
The volume of the sphere portion is 75.266.
The combined volume is 535.987.

The surface area of the cone portion is 59.537.
The surface area of the cylinder portion is 246.741.
The surface area of the sphere portion is 68.424.
The combined surface area is 374.702.
```

3 decimal places

Run your code with several different inputs and hand check the values to make sure your code is doing the calculations right.

Programming Style

An important part of programming is using proper programming style, formatting, and comments. At this point, the most important items are:

- Use descriptive variable names
- Vertical (proper use of blank line mostly) and horizontal spacing (putting spaces around operators, indentation, etc.)
    - Indent properly
    - Skip a line between each small section of your program, so that each section is separated by a blank line,
- Have a beginning comment that describes what the program is doing (just "Homework 2" is not good enough). See the main Canvas page for Coding Guidelines about what every program header needs to contain.
- Each section (i.e. logicl chunk of code) begins with a comment describing what that section does, so that the comments alone would provide an outline of the program (It's NOT good to follow every line of code with a comment.)
- Continue typing your code on the next line if it is too long. Too long is over 80 characters. In Code::Blocks, there is a column indicator at the bottom-middle-ish of the screen that tells you which column the cursor is at.

Submit Your Work

To submit your work to Web-CAT to be graded.