**PROBLEM**:

Write a program dealing with cylinders to compute the top area and the volume using an input data file and an output file. A file **lab4.c** is provided with some of the print statements in it.

* It is your responsibility to read the Power Point files:
  + C-2 Characters ControlStructures.pptx, #20-57 on the various Ifs and Switch
  + C-3 Loops.pptx, #1-24
  + Ask any needed questions in class

To get the files you need, first move to your class folder by typing: **cd csc60**

The following command will create a directory named **lab4** and put all the needed files into it below your csc60 directory.

Type: **cp -R /gaia/home/faculty/bielr/classfiles\_csc60/lab4 .**

Spaces needed: (1) After the **cp ↑** Don’t miss the space & dot.

(2) After the **-R**

(3) After the directory name at the end & before the dot.

After the files are in your account and you are still in **csc60**, you need to type: **chmod 755 lab4**

This will give permissions to the directory.

Next move into lab4 directory [**cd lab4**], and type: **chmod 644 lab4\*.\***

This will give permissions to the files.

Your new lab4 directory should now contain: lab4.c, lab4.dat, lab4sample.dat

**INPUT/OUTPUT DESCRIPTION**:

* The **test data** will be a file called **lab4sample.dat.** Use it to verify the correctness of your program. It has only two sets of data.
* The **final data** will be a file called **lab4.dat**. It has 4 sets of data.
* Each line or record of the file will consist of two numbers: the radius and the height of the cylinder.
* Use an *fscanf* statement in a *while* loop to repeatedly get each set of values.
* **The output** will be a file, **lab4.out.** The output of the sample data will be a model to follow.

**ALGORITHM DEVELOPMENT**:

Open the data file **lab4sample.dat** or **lab4.dat**

Do the appropriate error checking

Open the output file **lab4.out**

Do the appropriate error checking

Print your name and assignment (use fprintf)

Set the count to one

while ((fscanf(..., &radius, &height)) == 2) //Fill in the details for the “…”

| Compute the area and volume of the cylinder.

| print the cylinder count, radius, height, area, and volume as in the Defined Output Appearance.

|\_ Increment count by one

Close the two files

**FORMULAS**

The **area** of the top of the cylinder = M\_PI \* radius \* radius

The **volume** of the cylinder = M\_PI \* radius \* radius \* height

**VIEWING OUTPUT**

When you run the program, the whole thing is going to **lab4.out**. Open that file to see your output.

**REMINDERS**:

* Include your name and lab4 in your comment block, and in your output.
* All numeric variables are to be type **double,** except for the type **int** count.
* The input file name, which will be changed, ought to be in a #define statement. The file will come with two #define statements, for the test file and the final file. Just move the **//** from in front of one #define statement to the other #define statement.
* Most of the print and fprintf statements are included in lab4.c for you. The fprintf statements use variable names. Make your variable names agree….or change the variable names in fprintf statements. The file pointer name also needs to agree…either use mine or change to yours.
* FOR THE VALUE OF PI, use **M\_PI** from math.h (which we already have included).
* To compile, you will need to add **–lm** so math.h can be found. Type: **gcc –lm lab4.c**

**DEFINED OUTPUT APPEARANCE (*using lab4sample.out*)**:

Ruthann Biel. Lab 4.

Cylinder 1

The radius is: 5.000

The height is: 2.000

The top area is: 78.540

The volume is: 157.080

Cylinder 2

The radius is: 40.000

The height is: 15.000

The top area is: 5026.548

The volume is: 75398.224

**DATA FILES**:

There are two data files:

* lab4sample.dat – Use it to verify the correctness of your program. It has two sets of data.
* lab4.dat – It has 4 sets of data.

**PREPARE YOUR FILE FOR GRADING:**

*Make sure your program has been corrected to use* **lab4.dat** *and has been re-complied.*

When all is well and correct,

Type: **script StudentName\_lab4.txt** [Script will keep a log of your session.]

Type: **cat lab4.c** to display the code in your session.

Type: **gcc -lm lab4.c** to compile the code

Type: **a.out** to run the program

Type: **cat lab4.out** to show contents of the output file

Type: **exit** to leave the script session

**Turn in your completed session:**

Go to Canvas and turn in your script session (StudentName\_lab4.txt).