

CSCE 1030 Lab 2

General Guidelines: (for ALL of your programming assignments and labs)

- Use meaningful variable names.
- Use appropriate indentation.
- Use comments, especially for the header, variables, and blocks of code. *Please make sure that your submitted code has comments as points may be deducted for a lack of comments.*

Example Header:

```
/* Author:      Jane Doe (Jane.Doe@my.unt.edu)
   Date:
   Instructor:
   Description:  A small description in your own words
                  that describe what the program does. Any
                  additional flags needed to allow the
                  program to compile should also be placed
                  here.

*/
```

A. Declaring, Initializing, and Printing Variables

The following program contains missing code. Specifically, you are to fill in the missing code as directed below:

- Declare variable `var1` as an integer, `var2` as a character, and `var3` and `var4` as floating point numbers, but do not initialize these variables yet.
- Next, assign 30 as the value of `var1`.
- Then, assign `var2` to be the uppercase first letter of your last name. Now, assign `var3` to be the floating-point number 4.5.
- Finally, seeing as the `var4` contains the result of an arithmetic expression, print out the result of `var4` to the terminal. Be sure to print a newline after the result.

```

#include <iostream>
using namespace std;

int main( )
{
    /* Declare variables var1 - var4 as directed, but
       do not initialize these variables here */

    // Initialize variables var1 - var3 as directed

    // Compute result and store in var4 already done
    var4 = var3 + var2 % var1;

    /* Print out the value of var4 variable, as well
       as add a newline after */

    return 0;
}

```

Complete the missing code and save the file as **Lab2A.cpp**, making sure it compiles and check your result that is output to the screen, making sure the operation result is correct. Note that you will submit this file to Canvas.

B. Writing a Simple C++ Program

Write a small, but complete C++ program called **Lab2B.cpp** that calculates the volume of a sphere as follows:

- Prompt for and read in a floating-point number variable for the radius of the sphere in cm and store in the variable **radius**.
- Compute the volume using the formula:

$$volume = \frac{4}{3} * pi * radius^3$$

and save the result in the variable named **volume**. Assign the value 3.14159 to the variable **pi** for your calculations.

Use the C++ power function to compute the cube.

- Finally, print out a statement that provides the user the calculated surface area of the cylinder in cubic cm.

For example, my output might look like this:

```
$ ./a.out
```

```
Enter the radius of the sphere (in cm): 1
```

The volume of the sphere is:4.18879 cubic cm

Note that you will submit this file to Canvas.

C. Working with strings and type conversion

In this program, we will compute the bonus earned by an employee and display the result on the screen.

Write a C++ program called **Lab2C.cpp** that does the following:

- Using a suitable message, prompt for and read in the full name of the employee and save it in a `string` variable called **name**. Note that the name can have more than one word, so you need to use the `getline` function.
- Using a suitable message, prompt for and read in the salary of the employee and save it in a `double` variable called **salary**.
- Using a suitable message, prompt for and read in the percentage rate of bonus computation and save it in an `integer` variable called **rate**. Assume the user will enter a value between 5 and 15.
- Now compute the bonus earned by the employee by using the following formula:

$$bonus = salary * (rate/100)$$

and store the result in a `double` variable called **bonus**.

You must use the formula as it is provided: **DO NOT remove the parenthesis and DO NOT change 100 to a floating point number such as 100.0 or 100.00.**

- Finally display the result in the following format:

Employee_name earned a bonus of **XXXX.XX**

Employee_name must be replaced by the name entered by the user, and **XXXX.XX** must be replaced by the bonus computed by your program.

If you used the formula as it is, you probably got 0 as the bonus. Think why?

HINT: Integer Division and Typecasting.

Fix your code to get the correct bonus value by using the **static_cast** function.

Make sure you display exactly two numbers after the decimal point.

Before writing the code, you may want to write out the algorithm, or steps, on paper of how you propose to solve this problem. Working out the arithmetic by hand is also recommended to insure that you are outputting the correct values for each variable. Be sure to compile and run your code to make sure it outputs the expected output. Note that you will submit this file to Canvas.

Now that you have completed this lab, it's time to turn in your results. Once you've moved the files to your windows machine (using **winscp**), you may use the browser to submit them to Canvas for the **Lab2** dropbox.

You should submit the following files:

- **Lab2A.cpp**
- **Lab2B.cpp**
- **Lab2C.cpp**

Ask your TA to check your results before submission. The above two files *MUST* be submitted to Canvas by the end of your lab section.

Now that you've finished the lab, use any additional time to practice writing simple programs out of the textbook, lectures, or even ones you come up with on your own to gain some more experience.