Programming Project Report

Name: Jan Bobda
Date: 08/28/2018

Academic Integrity Statement: I pledge that I have neither given nor received unauthorized help on this programming assignment.

Problem Statement:

The goal of this programming assignment was to give me experience with the input and output commands of C++ as well as basic numerical calculations through calculating the volume, surface area, and metal mass of toy tops. For the program inputs, it took in the Radius of the larger sphere in centimeters (cm), the height of the connecting cylinder in centimeters (cm), and the radius of the smaller half sphere in centimeters (cm). Once it took in these inputs the program used the provided mathematical formulas to output the total volume of the top in centimeters cubed (cm^3), the total surface area of the top in centimeters squared (cm^2), and the mass of the top made out of Gold, Silver, and Steel in grams (g). The project required no error handling or sanitization of inputs.

Design:

In the design of this program, I first declared the local variables necessary for calculations at the beginning of the program. Then the program prompts the user for in the input, the text prompts specify that this input data should be entered as centimeters (cm) but because there is no requirement for error handling it is possible to get inaccurate results if the user enters bad inputs. After input is taken, the inputs are run through formulas ,which use a power function declared and defined at the top of the program, and the program calculates the necessary outputs. Then the total volume and surface area are printed out to the screen, and the mass in Gold, Silver, and Steel are calculated and also printed out to the screen. In this program no real data structures were necessary to have a functional design. Outside of the formulas provided in the assignment, no algorithms were used. The project design made it easy to debug and look over the program and make changes when necessary.

Implementation:

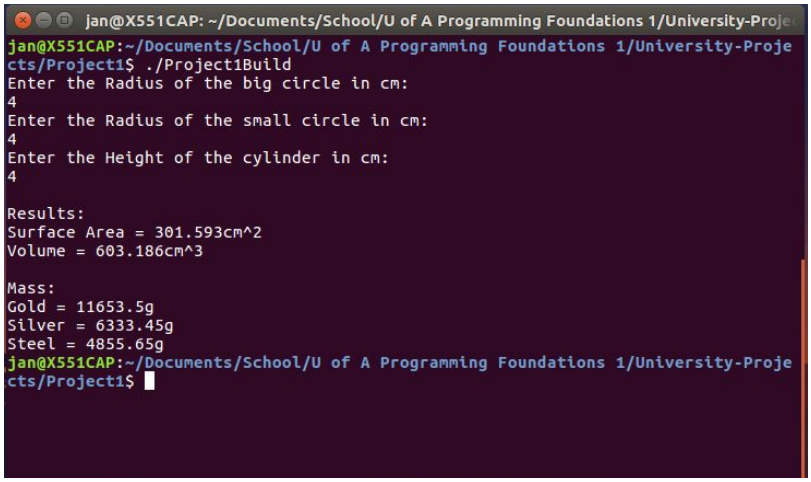
For the implementation I worked on this program incrementally, first starting with an empty main function to ensure that my programming environment was setup properly and adding more and more complexity, making sure that everything prior worked before moving on. I used the git version control system to make sure that I would always have a working version. I did not start with any sample code and had to write the program from scratch. I worked on the program for about an

hour, spending the first thirty minutes writing out the code incrementally, and then spent the other thirty minutes testing and debugging the program and making small adjustments to make sure everything worked right.

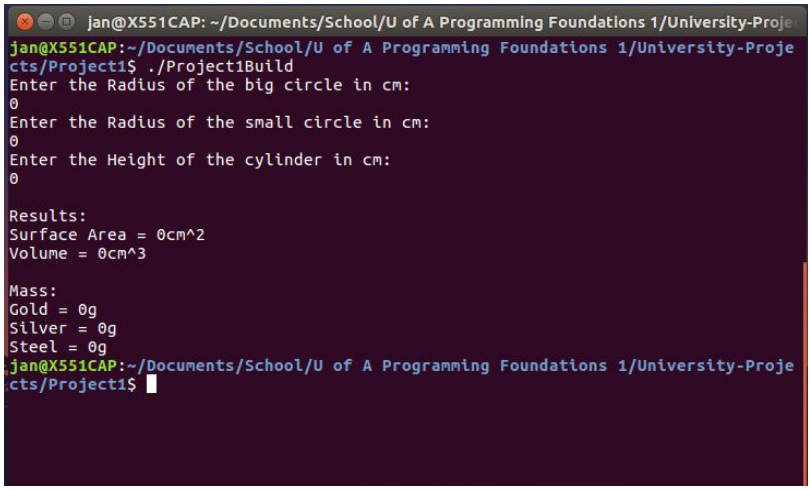
Testing:

I tested my program incrementally, I made sure that the program compiled first, and then placed varying inputs into the program to test it. Once the program was fully written I tried using various inputs such 4 cm for Radius 1, 4 cm for Radius 2, and 4 cm for the Cylinder height. I also tried some edge cases such as 0 cm for Radius 1, 0 cm for Radius 2, and 0 cm for the Cylinder height, and also floats as well as -4 for all the inputs. Lastly I tried some none integer inputs that I knew were going to break the program such as "Hello World." The program, from my testing, seemed to work properly but when faced with negative numbers, it gave back a negative volume and negative masses which are questionable results and the program took the non integer/float value and set all variables to zero

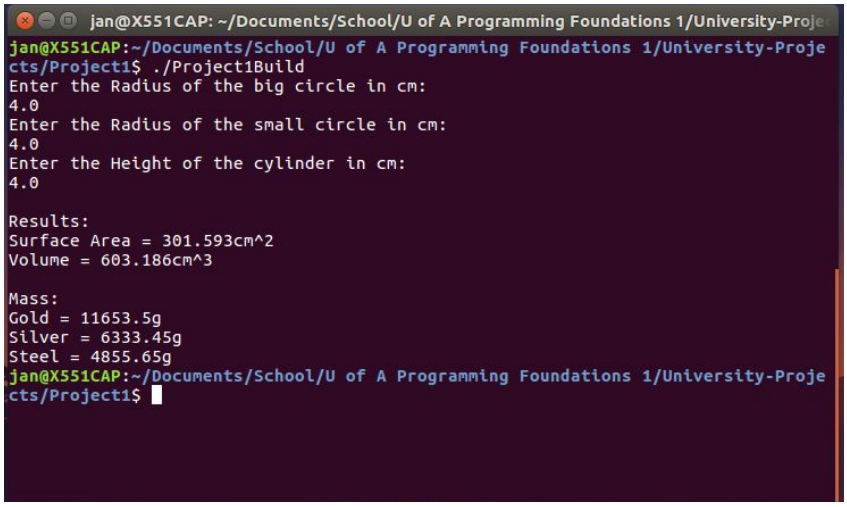
Radius 1 = 4 cm, Radius 2 = 4 cm, Height of Cylinder = 4 cm:



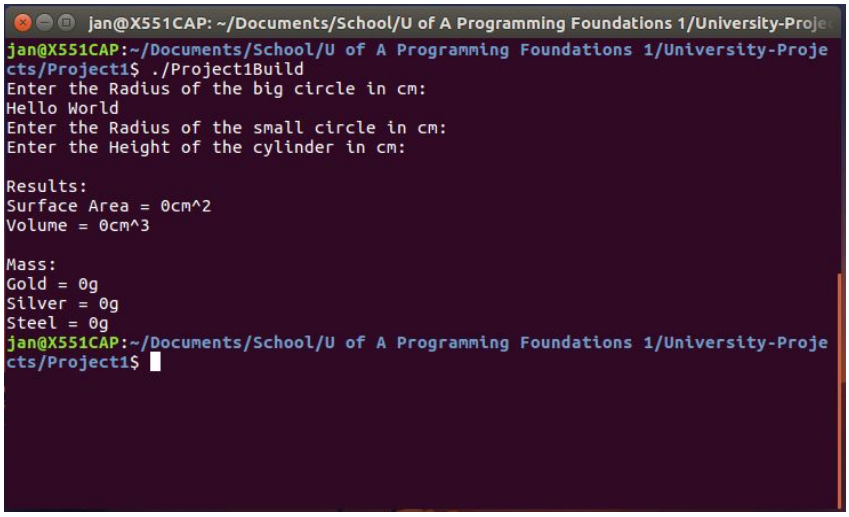
Radius 1 = 0 cm, Radius 2 = 0 cm, Height of Cylinder = 0 cm:



Radius 1 = 4.0 cm, Radius 2 = 4.0 cm, Height of Cylinder = 4.0 cm:



Radius 1 = “Hello World”, Radius 2 = N/A Height of Cylinder = N/A:



Conclusions:

Overall, the program seemed to do its job properly and I would consider the project a success. After the testing the output seemed reasonable in most cases when the program receives proper input. Next time I would have a better testing method in mind, throughout this project it seemed like my testing methodology was lacking and next time I would like to have it planned out more. The project took roughly around 1 hour to complete when factoring in the testing.