**Week 2 Practice Problems**

**#1.** Write a function that takes in a three-digit integer and returns the “flipped” version (that is, with its digits reversed). For example:

Enter a positive number less than 1000: **177**

That number flipped is 771

Hint: Represent the user’s input as an integer and use the arithmetic operators // and % to do it.

Make sure it works if the user enters a one or two digit integer. For example, 37 flipped is 730, and 8 flipped is 800.

**#2.** Write a function that asks a user for a positive integer, and then prints the minimum number of quarters, dimes, nickels, and pennies needed to make up that amount. For example:

Enter a number of cents: **67**

2 quarter(s), 1 dime(s), 1 nickel(s), 2 penny(ies)

In case you are not familiar with Canadian currency:

* a quarter is 25 cents
* a dime is 10 cents
* a nickel is 5 cents
* a penny is 1 cent

**#3.** Write a program that asks a user for a floating point number and that then prints two values: the number truncated to the first decimal place and the number rounded to the first decimal place. For example:

Enter a number: **4.158**

Truncated to one decimal place: 4.1

Rounded to one decimal place: 4.2

Hint: Have a look at the math module.

**#4.** Write a little trigonometry program. Ask a user for an angle, specified in degrees. Then print the sine, cosine, and tangent of that angle. For example:

Enter an angle: **60.**

sin(60.00) is 0.866025

cos(60.00) is 0.500000

tan(60.00) is 1.732051

**#5**. Write a function that takes a positive integer input less than 100000 and returns an integer corresponding to the number of digits in the number. Use this function in a program that prompts the user for an integer and prints the number of digits in the user’s input.

Note: there are a number of ways to solve this problem. It is pretty easy if you do it with a string. You should also try it directly with an integer. Don't use an if-statement (that is for next week).