CS 161 Lab Activity
More on Functions | Boolean Operations

Part One: More Function Problems

Write one function for each of the following. You can test each with a dummy program that just calls the function with the necessary number of arguments.

1) Convert Fahrenheit to Centigrade. Input is temperature in degrees Fahrenheit and the function returns the equivalent centigrade temperature. (To test, make sure that 32 F is 0 C and 212 F is 100 C).

2) Compute calories for a meal. Input is grams of protein, grams of carbohydrate and grams of fat. The function returns total calories. Protein and carbohydrate are 4 calories per gram, fat is 9 calories per gram.

3) Given a string that describes the desired numeric input, this function asks the user to enter the number (takes the input as a string) and verifies that each character is numeric. If the value is not numeric, the functions asks again. This requires to levels of iteration—one that keeps asking for a number until a valid number is entered, and one that tests each value of the entered string. A tip—the sentinel value for the outer loop will need to be handled so that it is true before the loop starts but toggles to false before each test of the input values.

Part Two: Boolean Operations

Here is a little code to illustrate how **and** and **or** work. You should be able to copy/paste. Try running this code with a variety of input values and studey the output for each. Make sure you understand why it works the way it does.

```
def testBoolean():

     first = input ("Enter first number>> ")
     second = input ("Enter second number>> ")
     third = input ("Enter third number>> ")
     fourth = input ("Enter fourth number>> ")

     if first == second and first == third and first == fourth:
          print "They're all the same"
     else:
          print "They are not all the same"

     if first == second or first == third or first == fourth:
          print "The first is the same as one of the others"
     else:
          print "First is different from all the others"

testBoolean()
```

Another challenge for you. You have seen this question before: how can we determine if a year is a leap year? If it is divisible by 400 it is, otherwise if it divisible by 4 but not divisible by 100 it is. "Divisible" can be written using the mod operation—"Is x divisible by six?" translates to **if x%6 == 0**.

We can write this complicated as a nested if structure. Try doing this and test your creation. We can also write it as a single if structure using Boolean operations. Try doing this.

A function that accepts the year as an argument and returns true if that year is a leap year and otherwise returns false is handy. Write this function.