Homework 2

Due Date: April 14, 2009 at 5:00PM **Points**: 100

Written Exercises

1. (*4 points each*) Copy the program `avg2.py` on page 38 of the text, and do the following:
   a. Put a dollar sign "$" immediately before and after each variable
   b. Put a carrot "^" immediately before and after each expression
   c. Put a comment at the end of each line indicating the type of statement on that line (output, assignment, input, loop, etc.)

   [*text*, §2.9, Discussion problem 2, modified]
2. (*4 points each*) Show the output from the following fragments:
   a.
   ```
   for i in range(5):
       print i * i
   ```
   b.
   ```
   for d in [3,1,4,1,5]:
       print d,
   ```
   c.
   ```
   for i in range(4):
       print "Hello"
   ```
   d.
   ```
   for i in range(5):
       print i, 2**i
   ```

   [*text*, §2.9, Discussion problem 4]

Programming Exercises

Remember to turn in your error logs. For problem 5, you must also turn in the refinement file.

3. (*20 points*) Modify the `convert.py` program in Section 2.2 with a loop so that it executes 5 times before quitting (i.e., it converts 5 temperatures in a row). Turn in your program in the file `convert2.py`.
   [*text*, §2.9, Programming Exercises problem 3]
4. (*20 points*) As an alternative to APR, the interest accrued on an account is often described in terms of a nominal rate and the number of compounding periods. For example, if the interest rate is 3% and the interest is compounded quarterly, the account actually earns ¾% interest every 3 months.

   Modify the `futval.py` program to use this method of entering the interest rate. The program should prompt the user for the yearly rate (`rate`) and the number of times that the interest is compounded each year (periods). To compute the value in 10 years, the program will loop `10*periods` times and accrue `rate/period` interest on each iteration.
   [*text*, §2.9, Programming Exercises problem 7]
5. (*32 points*) Write a program to compute the sum of the first *n* cubes (that is, each number from 1 to *n* raised to the third power, and the powers summed). Prompt the user to input *n*. You may assume that the user will enter a non-negative number. Turn in your program in the file `sumcubes.py`.

Extra Credit (Programming)

Remember to turn in your error logs. For problem 6, you must also turn in the refinement file.

6. Do the following:
   a. (*25 points*) Write a program to compute the square of the sum of the first *n* numbers. Prompt the user to input *n*. You may assume that the user will enter a non-negative number. Turn in your program in the file `sumnsq.py`.
   b. (*3 points*) What do you notice about the output from the program `sumcubes.py` and `sumnsq.py` when they are given the same *n*?