CSE 231 Fall 2007

Programming Project #1

Assignment Overview
This assignment involves coding and testing of a program based on the "Hello World" program from the first lab.

The basic design of the first programs that you construct in this class consists of a prompt for information, receiving information, processing that information then producing a display of the results.

This assignment is worth 10 points (1% of course grade), and must be completed before 11:59 PM on Tuesday, September 4th. Projects are typically due on Mondays, but this one is an exception because of the Labor Day Weekend.

Background
This programming project will use the `raw_input` and `print` functions along with some simple mathematics manipulation. The important part of the project is to learn the skills needed to access the class web site to download a project description, create a new program in Python and finally to hand it in.

Program Specifications
Your program will prompt the user for two integer numbers (numbers without decimal points). It will next display the following output:

- First line: the sum of the first and second number
- Second line: the result of subtracting the second number from the first
- Third line: the product of the first and the second number
- Fourth line: the integer division of the first number by the second number, followed by the remainder from dividing the first number by the second number

Deliverables
proj01.py -- your source code solution (remember to include your section, the date, project number and comments).

1. Please be sure to use the specified file name, i.e. "proj01.py"
2. Save a copy of your file in your CS account disk space.
3. Electronically submit a copy of the file.

Assignment Notes:
To input the numbers it is necessary to use the `raw_input` function. The `raw_input` function takes a string, a sequence of characters between quotes, as a prompt to print to the user. It then waits until the user types a response, terminated by the user typing the Enter key. A string, again as a sequence of characters, is returned.

The returned string must be converted to a number. Since in this assignment we are strictly working with integers, a string is converted to an integer using the `int` function. The `int` function takes as an argument a single string and returns the integer the string represents. A typical interaction would be something like:

```
numStr = raw_input('Please enter a number: ')
intVar = int(numStr)
```

`print` is a command that will print on the output window any combination of variables, values and strings. Each item to be printed must be separated from other items by a comma. All the items will be printed together, followed by a new line. For example:

```
billsInt = 3
print 'The number ',billsInt,' times two is ', billsInt*2
```

This command has 4 items to print: a string ('The number '), the value in the variable billsInt (3), another string (' times two is ') and the result of an expression (6). What it will print is:

```
The number 3 times two is 6
```

Look at the program numberInput.py in the proj01 directory as an example of using `raw_input, print` and `int` .

Once converted to numbers, the operations on these numbers are, respectively: **+** (sum), **-** (difference), ***** (product), / (division) and **%** (remainder). The last two deserve special comment.

In Python, if an integer is divided by another integer, the result is an integer. Thus the result of **6/4** is **1** (not 1.5). That is, the "/" operation results in the integer **quotient**. The result of **6%4** is the integer remainder of the division, thus **2** (6 divided by 4 is 1 with a remainder of 2). *Play around with the quotient and remainder operators in the Python shell window until you get comfortable with how it works*.

To clarify the problem specifications, we provide at the end of this document a snapshot of interaction with the already written program.

Getting Started

1. Using IDLE create a new program.
2. If you are in a CS lab, select the H: drive as the location to store your file
3. Save the name of the project: proj01.py
4. Using the example from numberInput.py, write the code. Track down any errors (shouldn't be any at first).
5. Run the program
6. Use the web site to hand in the program (to make sure you can do it)
7. Edit the program
8. Now you enter a cycle of edit-run to incrementally develop your program.

9. Hand in your final version.

Questions for you to consider (not hand in)

1. What happens when you try to divide by zero when you run your program?
2. What happens when you multiply two VERY LARGE integers?
3. What happens when you enter a letter instead of a number at the prompt?

Sample Interaction