CS 271 Computer Architecture and Assembly Language

Programming Assignment #1

Due the Sunday at the end of the 2nd Week of the Course (11:59 PM)

Submit at http://engr.oregonstate.edu/teach before midnight

Objectives:

1. Introduction to *MASM* assembly language
2. Defining variables (integer and string)
3. Using library procedures for I/O
4. Integer arithmetic

Description:

Write and test a MASM program to perform the following tasks:

1. Display your name and program title on the output screen.
2. Display instructions for the user.
3. Prompt the user to enter two numbers.
4. Calculate the sum, difference, product, (integer) quotient and remainder of the numbers.
5. Display a terminating message.

Requirements:

1. The main procedure must be divided into sections:
   - introduction
   - get the data
   - calculate the required values
   - display the results
   - say goodbye
2. The results of calculations must be stored in named variables before being displayed.
3. The program must be fully documented. This includes a complete header block for identification, description, etc., and a comment outline to explain each section of code.
4. Submit your text code file (*.asm*) by the due date. The submission site can be found at **http://engr.oregonstate.edu/teach** .

Notes:

1. A program shell (template) is available on the course website.
2. You are not required to handle negative input or negative results.
3. You may submit only one of your programs up to 48 hours late without penalty. Try not to use this on the first program. A second late submission will not be accepted.
4. To create, assemble, run, debug, and modify your program, follow the instructions at http://www.kipirvine.com/asm/gettingStartedVS2010/index.htm.
5. Find the assembly language instruction syntax in the textbook.
6. Find help on using Irvine library procedures in in the textbook.

Example execution (user input is in ***italics*****)**:

```
          Elementary Arithmetic     by Wile E. Coyote

Enter 2 numbers, and I'll show you the sum, difference,
product, quotient, and remainder.

First number: 37
Second number: 5

37 + 5 = 42
37 - 5 = 32
37 x 5 = 185
37 ÷ 5 = 7 remainder 2

Impressed?  Bye!
```

Extra-credit options (original definition must be fulfilled):

1. Repeat until the user chooses to quit.
2. Validate the second number to be less than the first.
3. Calculate and display the quotient as a floating-point number, rounded to the nearest .001.
4. Do something astoundingly creative.