CSE 231, Fall 2007

Programming Project 03

Assignment Overview

This assignment is worth 30 points (3.0% of the course grade) and must be completed and turned in before 11:59 PM on Monday, September 17, 2007. The purpose of this project is to familiarize you with the use of Boolean logic, branching statements and loops. For this assignment, you will create a program that is a simple calculator.

Background

You calculator will perform addition, subtraction, multiplication and division. It will accept floating-point values, and it will check for division by zero.

Input will be in the form:

operand operator operand

where operand can be any Real number and operator is +, -, *, or /.

The operator **must** be separated from the operands by at least one space, i.e. "4+3" will cause Python to generate an error. We cannot protect ourselves from that error (yet) so we will simply assume that the user will not make that error when entering values.

When you hit the Enter key, the result will be printed. If an illegal operand is used or division by zero is attempted, an error message will be printed, but the program will continue asking for a new expression to evaluate.

After a result is printed your program will ask if you wish to perform another calculation. If 'y' or 'Y' or 'yes' or 'Yes' or 'YES' is entered, another calculation can be performed, otherwise the program will quit.

Program Specifications

This project requires you to use:

- raw_input to prompt the user
- print to print results
- at least one branching mechanism (if statement)
- at least one loop (while loop)
- Boolean logic

Deliverables

You must use handin to turn in a file called **proj03.py** – this is your source code solution; be sure to include your section, the date, the project number and comments describing your code. Please be sure to use the specified file name, and save a copy of your proj03.py file to your H drive as a backup.

Assignment Notes

1. A sample implementation called proj03.pyc will be available for you to run so that you may see how your program should run.
2. `raw_input` should be used for prompting. It returns a string containing the user's choice.
3. The string method `lower()` may be useful to make the choice input lower case for easier checking; `'ABC'.lower() => 'abc'` Its use is not required.
4. The membership operator `in` could be useful as well for determining what a correct response is. It use is not required.
5. In the project directory is a Python program that demonstrates a `while` loop handling input. We suggest that you copy this file and use it as the starting point for your program. Run the program and play with input until you are confident that you understand what it is doing.

Getting Started

1. Do all the standard startup things. Create a new file called proj03.py. Put your comments in at the top, save it.
2. Now you need to break the problem down into parts. Read the description and identify the subtasks that need to be solved. In this project we have started that process by providing a starting program. Using comments mark where you will add subtasks to the file.
3. A logical subtask would be to do the actual calculations. Assuming that all input is perfectly correct, draw up the code to take the input and perform the operation.
4. Once you have your program handling calculations you can add error checking, e.g. division by zero and correct operators.
5. Remember; add a subtask to your program followed by saving the file and running it. Developing code one subtask at a time will make debugging the program easier.