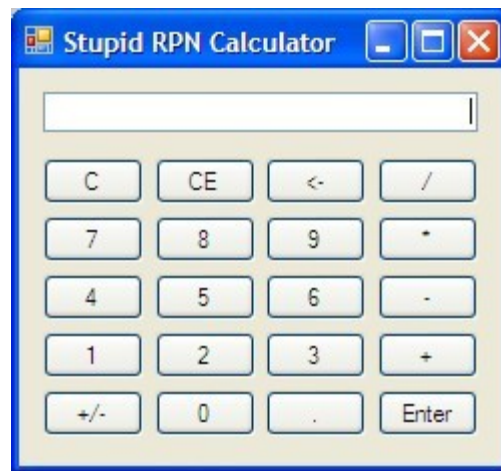


CS 205 - Programming for the Sciences Spring 2008 - In-class Exercise for 01/15/08

Today's exercise is to complete a very simple (and stupid) RPN calculator. For this exercise, you are given a C# project with a GUI already created. You are to write the C# code to provide the functionality.

Use a Web browser go to the course webpage <http://csserver.evansville.edu/~hwang/s08-courses/cs205.html>. Under today's date, right-click the link to the file StupidRPNCalculator.zip. Save this compressed folder to your network drive. From My Computer, browse to the compressed folder. Right-click on the compressed folder and select Extract All. Go through the steps of the Extraction Wizard ending with a window with the extracted solution folder. Double-click into the folder, then double-click on StupidRPNCalculator.sln (the Visual Studio solution file). This will launch Visual Studio with the solution loaded.

The GUI design for the program has been completed. To see it, right-click on Form1.cs in the Solution explorer window and select View Design. It should look like the following:



Currently, there is no handler code for any of the buttons.

This GUI is intended to be used as an interface to an RPN (Reverse Polish Notation) calculator. In RPN, both operands are specified first, then the operator to applied to the operands. For example, the expression $3 + 4$, would be written as $3\ 4\ +$ in RPN. For this calculator, to indicated the end of the first operand, the Enter button is used. So the calculator button sequence for the previous expression would be 3, Enter, 4, +. This calculator is stupid, since it can only handle expressions with two operands, though it can accumulate results. More on this during class.

Some GUI notes:

- The buttons were made exactly the same size by explicitly setting the Size property (length, width) of each button. They were sized to fill a form that is wide enough for the form titlebar text to be seen in its entirety.
- The textbox TextAlign property was set to Right. (Default is left.)

Assignment

(10 points) Working in pairs, complete this program by writing handlers for each of the buttons in the following order:

- The number and decimal point buttons that add digits to the end of the current operand
- The operator and Enter buttons that compute a result value and make it the current operand
- The +/- button that negates the current operand
- The clear (C) and clear error (CE) buttons. The clear button is intended to reset the calculator to its initial configuration. The clear error button is intended to clear only the current operand.
- The backspace (<-) button.

We will work through the implementations of these handlers during class. When you are finished, demonstrate your program to an instructor.

Extension to consider

Add one memory location and buttons for the following operations: the current operand is stored into the memory location, the stored value is retrieved and made the current operand, the memory location can be cleared (set to 0), and the current operand can be added to the stored value. Often the buttons for these operations are labeled MS, MR, MC, and M+, respectively.