A290/A590
Tools for Computing
Windows Interface Design and Programming with C# and .NET.

Homework Project 3

DUE Saturday, December 3, 2016 by 11:59:00.00PM (SHARP)
Submit ZIPPED file "A3.zip" of entire Project Folder to the "Homework Project 3" Assignment in the Canvas "Homework Projects" Group

Preliminary Information:

Your goal is to create a fairly simple "Reverse Polish" or Post Fix Calculator. The basic instructions for doing most of this are contained in various Example Guide files available via the Meeting Outlines on the website. We went through most of the basics of this project together, so adding the modifications is the only real "challenge" to this assignment. Finally, you will have to ZIP your project folder **[USE WinZip compatible tools]** into its own single ZIP file and submit your file to your Oncourse "Homework Programs" Dropbox by the deadline indicated above.

Part I. Design Goals - Details.

This is again a **very** crucial aspect of this last Assignment. You want to create a calculator application that is as "sleek" and "professional" in its appearance as possible, while at the same time reflecting your personal "unique" take on what it is to be a calculator. In addition, you want its use to be as intuitively obvious as you think possible. This means you need to give very careful consideration to (among other things): 1) the size of your form, 2) the size, appearance and placement of your function buttons, and 3) the size, appearance and placement of your data entry and result fields.

Part II. Technical Requirements (including Design Goals) [100 points]

- Address the Design goals stated in Part I, as well as any other considerations you think relevant.
- Inclusion of a README.TXT file with your project, **in the root project folder**, clearly explaining your design choices and how they are presented in your final applications. **This is a required component.**
- Proper Main Form.
- Reverse Polish/Post Fix Design, *i.e.*, no "=" sign. Two values entered and use of any function "key" generates result in "output" field.
- Two fields for data entry.
- Result field.
- 4 basic arithmetic functions, +, -, *, /.
- Appropriate check routine(s) to insure proper "reaction" to any entries that are NaN (Not a Number).
- ANY OTHER data entry checks or other checks you think are required to insure you are "taking care of your user."
- Some obvious control for clearing the data fields to start a new calculation.
- **At least four (4)** additional arithmetic/mathematical functions that can be applied to one or both of the values entered in your data field(s). Their function and use need to be **as intuitively obvious** as the 4 basic operators.
- A "Quit" button or other equally obvious control to gracefully exit the application.
- All required general information/comments at the head of the file.
- Appropriate comments within the body of the code to clearly explain what is being done and how.

Design goals addressed: 25 points
README.TXT: 10 points
General and specific comments: 10 points
Form and layout: 10 points
Data entry and result fields and proper manipulation of strings: 15 points
4 basics functions: 5 points
4 additional functions: 15 points
Error/exception handling: 10 points

Part III. Handing it in.

It should be clear that failure to successfully submit your ZIPPED [Please use WinZip compatible tools] file to Canvas or failing to meet the deadline will result in a score of zero (0). Partial credit will only be possible if you are unable to make the assignment work, but are successful in submitting your assignment and most or all of what you submit is correct. Make sure your file/folder ends up with a name that makes it clear what it is. If you have questions about this, ask them ASAP.