Homework #1: Hello World

Design a Windows Phone to fulfill the following specifications. The specifications are presented in a series of phases, however it is not necessary for you to complete each phase and submit them separately. Only the final phase must be submitted, this separation into phases is to provide stepping stones toward the final product.

Phase 1: You will develop a simple calculator application using TextBox, Button and TextBlock controls. All logic will be done in C#. Aesthetics are completely to your discretion, but basic functionality should include addition, subtraction, multiplication and division of numbers. This "Calculator app" should function similarly to most other calculator apps on cellphones, where there is an internally stored number that is modified by the different mathematical operations. See the provided solution for an example of how this functionality should by implemented.

Phase 2: Place your UI objects within a Pivot control, where one page will have functionality governed by C# code, and one page will have functionality governed by C++ code. This is easily done by copy-pasting your existing XAML code into a PivotItem, however note that when duplicating your UI for the C++ page, you will need to change any unique identifiers appropriately, to avoid name clashes.

Phase 3: Create a C++ component to perform calculations just as has been done in C# in Phase 1. Have a set of two independent Pivot Items, one with calculations performed by C#, and one with an identical set of UI controls but calculated by C++. Keep all state separated between the two, (e.g. my calculations in the C++ side should be independent of the calculations in the C# side). Because we have not covered conversion from strings to floating point numbers in class, you may do all your string conversion in C#, only dealing with floating point numbers in C++.

Note that you will have to have C# "wrapper code" to link the XAML button presses to your C++ functions, however all arithmetic logic should be performed by the C++ code when applicable. Further note that crashing your application may be possible. You should come up with some kind of reasonable solution to stop this from being possible; I am not dictating how to make this change, the implementation is up to you.

Submit a .zip file of all contents of your repository to the class dropbox.