C101 Homework #10 Due Wednesday 11/14/12

This homework will give you more practice in writing functions and also how numbers are read into a variable. You need to write a function that will read an unsigned integer into a variable of type *unsigned short int*. This will have a maximum value of 65535, and the function needs to take care of illegal numbers. **You can not use** "*cin >>*", inside the function.

The rules for numeric input are basically as follows:

1) skip all leading white spaces
2) first character found must be numeric else an error will occur
3) numeric characters are then processed one at a time and combine with number
4) processing stops when non-numeric found

We will follow these rules and also add error handling and overflow. If an illegal entry is made before a numeric than an error code of "1" will be sent back, if overflow occurs, that is number bigger then 65535, then error code of "2" will be sent back. If no error then "0" is sent back.

Make sure the main function will continue to loop until the user enters a “n” or “N” for NO, the main should test the error code returned from the function called “ReadInt” and display appropriate error messages or display the number if there is no error. Take care in designing the “ReadInt” function, it should be value returning and have a reference parameter. The function needs to process one character at a time from the input buffer and deal with it in a correct fashion. Once the number has been read in, then make sure the input buffer is empty, otherwise the loop in main may not work correct. I know this is not how the extraction works, but lets do it this way.

You do not need to turn in an algorithm with this assignment, but I would advise you to write one. And the debugger may prove helpful as well. You are basically rewriting the extraction operator as it works on integers.