**Institute of Technology Tralee**

**Ord/Hons BSc. in Computing with Specialism (Group B) - Year 1**

**Continuous Assessment #2**

**Date: 1/12/11**

**Time: 3 – 5 p.m.**

**Introduction to Programming**

**Instructions:** Attempt the following question. You should use the Just BASIC IDE for coding. When you are finished you must print out your code for correction.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Q1.**

A *palindrome* is a word that spells exactly the same backwards and forwards so, for example, the phrases

“Navan” and “Able was I ere I saw Elba”

are both examples of palindromes. Note that case does not matter when determining whether a piece of text constitutes a palindrome, as indicated in the examples above.

You must write a Just BASIC program that first of all reads in a user-supplied piece of text. This piece of text must initially be validated to ensure that it only contains letters of the alphabet (upper or lowercase) or space characters. As long as the text entered remains invalid, it should be rejected and the user prompted to re-enter. You can use **while-wend** loops here for validation purposes.

When the user has entered a valid piece of text, it must then be analysed to determine whether it constitutes a palindrome or not. A message should then be issued indicating whether or not this is the case. You can use any structures you like when coding this section.

The user should be asked to enter **exactly five** valid pieces of text in total (use a **do-while** loop for this process).

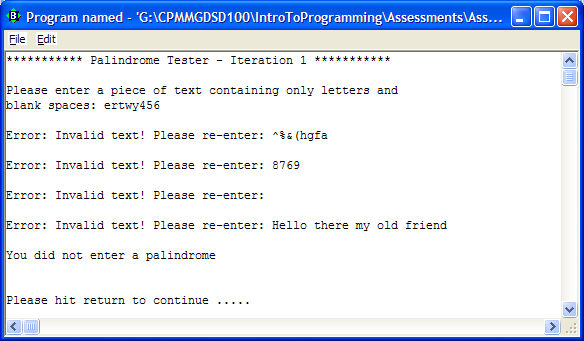
In coding your solution, you should find the **lower$**(), **asc**(), **mid$**(), **len**() functions to be particularly useful.

Using the test values as indicated in the screen shots below, the program should give you **exactly** the following output when it runs, including banners, blank lines, cleared screens etc.

Also note that there will be a few marks awarded for having a **meaningful comment at the top of the program** and for ensuring that your program **terminates properly** by handing back all the resources it has used to the system on completion.

**Sample Screen Shots**

**In this first iteration the user is prompted for text. As long as the text contains any characters other than letters or blanks (or if the user just hits return), the text will be rejected and the user asked continuously to re-enter until valid text is supplied. When the valid text gets supplied it is tested to see if it is a palindrome (it is not in this case) and then the program halts until the user hits return on the keyboard**



**Once the user hits return, the screen clears and the second iteration begins. The user is prompted for another piece of text and the validation process continues as before. Again, as soon as a valid piece of text is entered it is checked to see if it is a palindrome. It is in this case and again the program halts until the user hits return.**

