****
Proj_2G_Numerical_entry

A WAY OF DISPLAYING NUMBERS
The ability to illuminate any segment rather than just the vertical ones means that numbers can be displayed.

IT INTRODUCES:

1. Subroutine display_num_string ();
This uses subroutine I2C_Tx_any_segment() to display the digits 0 to 9

2. The #define statement.
This is used here to define each digit in terms of its segment letters. Each set of segment letters is known as a string and is stored in program memory in an array terminated in the null character '\0' or (0). When the compiler sees a digit name (i.e zero) it substitutes the address at which the first segment is stored.

3. The askii code
This is used to represent symbols in numerical form.
By incrementing the number from 32 to 126 most of the typewriter symbols from space to ~ are covered. For example the letters a to g are represented by the numbers 97 to 103.

4. The macro User_prompt: This repetitively sends a char (R) to the PC and pauses program execution until the user echoes it by pressing either R or r.

5. More on pointers:

Consider the statement const char* string_ptr = 0;
The memory location "string_ptr" is preceded by a "*".
This tells the compiler that it will be used to hold the address of data to be operated on rather than the data itself (in this case the address of the first segment used to define a digit).

Consider the statement display_num_string(string_ptr, digit_num);
It call subroutine display_num_string() passing the data contained in string_ptr and digit_num
However string_ptr contains an address and digit_num contains a number between 0 and 7

Consider the subroutine display_num_string (const char* s, int digit_num)
It provides a memory location for variable digit_num.
It does not provide memory for the variable s. Instead it uses the address stored in string_ptr.

6. The "continue" statement.
Note that in a loop such as "for" or "while" a "break" statement causes program execution to jump to the statement following the end of the loop.
The "continue" statement causes program execution to jump to the bottom of the loop from where it repeats the loop again. Note that "break" is also used in the "case" construct.

Note:
If the number 45 is entered the number 54 gets displayed. This is a common problem with displaying and printing numbers.