Assignment #10

Recursion

Write *recursive* definitions for each of the following Python functions, and for each function, include a clear and concise comment to describe its purpose. Use only the Python topics covered so far in class.

1. `Power( x, n )`

   The number 'x' raised to the power of the non-negative integer 'n'

   (do *not* use the Python operator '**' here)

```
Power(  2,  4 ) ⇒ 16
Power( -2,  3 ) ⇒ -8
Power(  7,  0 ) ⇒ 1
Power(  0,  0 ) ⇒ 1     (differs from mathematical definition)
Power(  0,  3 ) ⇒ 0
Power( 1.5, 2 ) ⇒ 2.25
```

2. `Range( lo, hi )`

   The list of integers from the integer 'lo' (inclusive) up to the integer 'hi' (exclusive)

   (do *not* use the builtin Python function 'range' here)

```
Range( 3, 7 ) ⇒ [ 3, 4, 5, 6 ]
Range( 3, 4 ) ⇒ [ 3 ]
Range( 3, 3 ) ⇒ [ ]
Range( 3, 1 ) ⇒ [ ]
```

3. `DigitCount( n )`

   The number of digits in the non-negative integer 'n'

   (recall that in Python, for example, `278 // 10 = 27`)

```
DigitCount( 278 ) ⇒ 3
DigitCount(  4  ) ⇒ 1
```

4. `Groups( s, k )`

   The list of strings of all groups of 'k' elements from string 's', where $0 \leq k \leq$ `len(s)`, and where elements in these strings occur in the same order as in 's'

```
Groups( "abcde", 2 ) ⇒
      [ "ab", "ac", "ad", "ae", "bc", "bd", "be", "cd", "ce", "de" ]
Groups( "abcde", 1 ) ⇒ [ "a", "b", "c", "d", "e" ]
Groups( "abcde", 0 ) ⇒ [ "" ]
Groups( "abcde", 5 ) ⇒ [ "abcde" ]
```

Program Submission:

Store the function definitions in a file named 'a10.py', and turn it in for grading by typing:

```
submit-cs1117 a10.py
```

Due Date: Fri Nov 20, 11:00am