Review for Test #3 -- NOT USED this quarter

The coverage for Test #3 focuses on everything we have done since Test #2, especially working with files and manipulating lists. You could see nested loops, which did not appear on Test #2.

[Can you think of a topic that should be listed below?  Post a message in the Discussion Forum with suggestions!]

Topic List:

* nested loops
* problem-solving with loops
  + counters, accumulators, flags
* functions that "return" a value
  + how to write them ('return' statements)
  + how to call them
* working with files
  + file variables -- data type 'file'
    - software objects and method calls
  + 'open' statements -- the open() function
    - file access modes: 'r', 'w' and 'a'
  + reading data in from a file
    - read() method
    - readline() method
    - standard pattern for reading a file line-by-line using readline() and a 'while' loop
    - working with string values -- using rstrip() as needed
    - working with numbers -- using int() or float() as needed
  + writing data out to a file
    - write() method, and how its usage differs from the print() function
  + close() method
* working with lists
  + sequences in general
  + terminology
    - list (array)
    - item (element)
    - index (subscript)
  + accessing one element in a list using an index -- [ ]
    - legal index values and the IndexError
  + changing the value of a list element -- lists are mutable
  + data type of the list variable (list) vs. data type of one element in a list vs. data type of an index
  + len() function
  + iterating through a list
    - standard 'for' loop pattern to iterate through one element value at a time
    - standard 'for' loop pattern to iterate through one legal INDEX at a time
    - variations on the "indexed" loop pattern
  + list algorithms
    - counts
    - sums/averages
    - highest/lowest
    - simple (sequential) search ('contains')
    - "index of the first occurrence" search ('index\_of')
    - sifting
  + using a list as a function parameter
    - object reference passed -- function can change the contents of the list
  + using a list as a function return value
  + list accumulators
    - creating an empty list
    - using the append() method
  + slices
    - a slice always creates (evaluates to) a new list
    - slice syntax
    - default values for start, "stop before" and step values
  + list copy vs. alias
  + other list operations
    - repetition operator (\*)
    - concatenation (+)
    - in and not in operators
    - other built-in functions that work with lists: sum(), max(), min()
* working with strings as sequences of characters
  + len() function
  + strings are immutable
  + accessing one character in a string using an index -- [ ]
    - legal index values and the IndexError
  + standard iteration patterns -- same as for lists
  + string accumulators
  + slices
  + other string operations
    - repetition operator (\*)
    - concatenation (+)
    - in and not in operators (used with individual characters or substrings)
* methods of a string object
  + isalpha(), isdigit(), isalnum(), isupper(), islower(), isspace()
  + upper(), lower(), lstrip(), rstrip(), strip(), lstrip(ch), rstrip(ch), strip(ch)
  + startswith(), endswith(), find(), replace(), split()