Programming Language Learning Series
Mastery of Python Language
(Interview Questions/Assignment-Procedural Style)

Q1: Write a function to compute 1/2+2/3+3/4+...+n/n+1 with a given n (n>0).

Q2: Write a function to find the sum of all the multiples of 3 or 5 below 1000.

Q3: A palindromic number reads the same both ways. The largest palindrome made from the product of two 2-digit numbers is 9009 = 91 × 99. Write a function to find the largest palindrome made from the product of two 3-digit numbers.

Q4: We count 35 heads and 94 legs among the chickens and rabbits in a farm. Write a python function that returns how many rabbits and how many chickens do we have.

Q5: Given a text file as input, we are interested to computing the following text analytics on that input:

- Compute the number of words in the given file
- Find the 10 most frequent words in the given file
- Find the number of times a given word appears in the file

Assuming that we want to develop a solution for the required text analytics using procedural abstractions. Which abstraction do you prefer and why?

Procedural Abstractions-I:

```
def wordcount(filename, word):
"""Return the count of the given word in the file"""

def top10(filename):
"""Return a list of the top 10 most frequent words in the
file"""

def totalwords(filename):
"""Return the total number of words in the file"""
```

Client Code:

```
print(wordcount("test.txt", "algorithmica")
print(top10("test.txt"))
print(totalwords("test.txt"))
```

Procedural Abstractions-II:

```
def read_words(filename):
"""Return a list of words in the file"""

def wordcount(wordlist, word):
"""Returns a pair (count, allcounts). count is the number of
occurrences of the given word and allcounts is a dictionary
from words to counts."""

def top10(wordcounts):
"""Return a list of the top 10 most frequent words in the
dictionary, in order."""

def totalwords(wordlist):
"""Return the total number of words in the list"""
```

Client Code:

```
words = read_words("test.txt")
(cnt, allcounts) = wordcount(words, "algorithmica")
print(cnt)
print(top10(allcounts))
print(totalwords(words))
```

Procedural Abstractions-III:

```
def read_words(filename):
"""Return a dictionary mapping each word in filename to its
frequency in the file"""

def wordcount(wordcounts, word):
"""Return the count of the given word in the dictionary."""

def top10(wordcounts):
"""Return a list of the top 10 most frequent words in the
dictionary, in order"""

def totalwords(wordcounts):
"""Return the total number of words used to create the
dictionary""
```

Programming Language Learning Series
Mastery of Python Language
(Interview Questions/Assignment-Procedural Style)

Client Code:

```
wordcounts = read_words("test.txt")
print(wordcount(wordcounts, "algorithmica")
print(top10(wordcounts))
print(totalwords(wordcounts))
```

decorators: @static_method, @lru_cache, @class_method

monkey patching code