Help

WORD SCORES (10/10 points)

The first step is to implement some code that allows us to calculate the score for a single word. The function `getWordScore` should accept as input a string of lowercase letters (a *word*) and return the integer score for that word, using the game's scoring rules.

A Reminder of the Scoring Rules

Scoring

- The score for the hand is the sum of the scores for each word formed.
- The score for a word is the sum of the points for letters in the word, multiplied by the length of the word, plus 50 points if all *n* letters are used on the first word created.
- Letters are scored as in Scrabble; A is worth 1, B is worth 3, C is worth 3, D is worth 2, E is worth 1, and so on. We have defined the dictionary `SCRABBLE_LETTER_VALUES` that maps each lowercase letter to its Scrabble letter value.
- For example, 'weed' would be worth 32 points ((4+1+1+2) for the four letters, then multiply by `len('weed')` to get (4+1+1+2)*4 = 32). Be sure to check that the hand actually has 1 'w', 2 'e's, and 1 'd' before scoring the word!
- As another example, if *n*=7 and you make the word 'waybill' on the first try, it would be worth 155 points (the base score for 'waybill' is (4+1+4+3+1+1+1)*7=105, plus an additional 50 point bonus for using all *n* letters).

HINTS

- You may assume that the input `word` is always either a string of lowercase letters, or the empty string `""`.
- You will want to use the `SCRABBLE_LETTER_VALUES` dictionary defined at the top of `ps4a.py`. You should not change its value.
- Do **not** assume that there are always 7 letters in a hand! The parameter `n` is the number of letters required for a bonus score (the maximum number of letters in the hand). Our goal is to keep the code modular - if you want to try playing your word game with *n=10* or *n=4*, you will be able to do it by simply changing the value of `HAND_SIZE`!
- **Testing:** If this function is implemented properly, and you run `test_ps4a.py`, you should see that the `test_getWordScore()` tests pass. Also test your implementation of `getWordScore`, using some reasonable English words.

Fill in the code for `getWordScore` in `ps4a.py` and be sure you've passed the appropriate tests in `test_ps4a.py` before pasting your function definition here.

Canopy specific instructions: If you modify code in `ps4a.py` go to

```
Run -> Restart Kernel (or hit the CTRL with the dot on your keyboard)
```

before running `test_ps4a.py`. **You have to do this every time you modify the file** `ps4a.py` **and want to run the file** `test_ps4a.py`, otherwise changes to the former will not be incorporated in the latter.

```
def getWordScore(word,n):


    word = word.lower()

    leng = len(word)

    b = []

    for i in range(len(word)):
        b.append(word[i])
        i +=1
    sum = 0
    total = 0
    for i in range(leng):
        for k, v in SCRABBLE_LETTER_VALUES.iteritems():
```

Correct

Test results

Hide output

CORRECT

Test 1

Function call: getWordScore('', 10)

Output:

0

Test 2

Function call: getWordScore(qi, 7)

Output:

22

Test 3

Function call: getWordScore(was, 7)

Output:

18

Test 4

Function call: getWordScore(outgnaw, 7)

Output:

127

Test 5

Function call: getWordScore(triplet, 7)

Output:

```
113
```

Test 6

Function call: getWordScore(triplet, 8)

Output:

```
63
```

Test 7

Function call: getWordScore(dogs, 4)

Output:

```
74
```

Test 8

Function call: getWordScore(cats, 7)

Output:

24

Test 9

Function call: getWordScore(kids, 5)

Output:

36

Test 10

Function call: getWordScore(onomatopoeia, 12)

Output:

242

Hide output

Check Save *You have used 14 of 30 submissions*

Show Discussion

New Post

EdX offers interactive online classes and MOOCs from the world's best universities. Online courses from MITx, HarvardX, BerkeleyX, UTx and many other universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, math, medicine, music, philosophy, physics, science, statistics and more. EdX is a non-profit online initiative created by founding partners Harvard and MIT.

Terms of Service and Honor Code

Privacy Policy (Revised 4/16/2014)

About edX

About

News

Contact

FAQ

edX Blog

Donate to edX

Jobs at edX

Follow Us

Twitter

Facebook

Meetup

LinkedIn

Google+