On Thursday, February 16th at 6:00AM EST, UTC-5, we will be conducting a brief database maintenance. The event should last about 5 minutes.

MITx: 6.00.1x Introduction to Computer Science and Programming U..

Help

Bookmarks

- Welcome to the edX Platform
- Entrance Survey
- Download Python and Get Motivated!
- Week 1: Python Basics
- Week 2: Simple Programs
- Week 3: Structured Types
- **Week 4: Good Programming Practices**

7. Testing and Debugging
Finger Exercises

8. Exceptions and Assertions
Finger Exercises

Problem Set 4

Week 4: Good Programming Practices > Problem Set 4 > Computer Choosing a Word and Playing a Hand

Computer Choosing a Word and Playing a Hand

Bookmark this page

Part B is dependent on your functions from `ps4a.py`, so be sure to complete `ps4a.py` before working on `ps4b.py`

Now that you have completed your word game code, you decide that you would like to enable your computer (SkyNet) to play the game (your hidden agenda is to prove once and for all that computers are inferior to human intellect!) In this part, you will be able to compare how you as a user succeed in the game compared to the computer's performance.

You should look at the following two functions: `compChooseWord` and `compPlayHand`, before moving on to Problem 7 on the next page.

compChooseWord

If you follow the pseudocode for `compChooseWord`, you'll see that the code creates a computer player that is legal, but not always the best. Try to walk through and understand our implementation.

A Note On Runtime: You may notice that things run a bit slowly when the computer plays. This is to be expected - the `wordList` has 83667 words, after all!

Test Cases to Understand the Code:

Problem Set due Feb 23, 2017 15:30 PST

- Midterm Exam
- Week 5: Object Oriented Programming
- Sandbox

```
>>> compChooseWord({'a': 1, 'p': 2, 's': 1, 'e': 1, 'l':
1}, wordList, 6)
appels
>>> compChooseWord({'a': 2, 'c': 1, 'b': 1, 't': 1},
wordList, 5)
acta
>>> compChooseWord({'a': 2, 'e': 2, 'i': 2, 'm': 2, 'n':
2, 't': 2}, wordList, 12)
immanent
>>> compChooseWord({'x': 2, 'z': 2, 'q': 2, 'n': 2, 't':
2}, wordList, 12)
None
```

compPlayHand

Now that we have the ability to let the computer choose a word, we need to set up a function to allow the computer to play a hand - in a manner very similar to Part A's `playHand` function. This function allows the computer to play a given hand and is very similar to the earlier version in which a user selected the word, although deciding when it is done playing a particular hand is different.

Test Cases to Understand the Code:

`compPlayHand({'a': 1, 'p': 2, 's': 1, 'e': 1, 'l': 1}, wordList, 6)`

```
Current Hand: a p p s e l
"appels" earned 110 points. Total: 110 points
Total score: 110 points.
```

`compPlayHand({'a': 2, 'c': 1, 'b': 1, 't': 1}, wordList, 5)` Current Hand: a a c b t "acta" earned 24 points. Total: 24 points Current Hand: b Total score: 24 points. `compPlayHand({'a': 2, 'e': 2, 'i': 2, 'm': 2, 'n': 2, 't': 2}, wordList, 12)`

```
Current Hand: a a e e i i m m n n t t
"immanent" earned 96 points. Total: 96 points

Current Hand: a e t i
"ait" earned 9 points. Total: 105 points

Current Hand: e
Total score: 105 points.
```

Computer Choosing a Word and Playing

Topic: Problem Set 4 / Computer Choosing a Word and Playing

Show Discussion

POWERED BY OPENedX

Download on the App Store

ANDROID APP ON Google play