

Midterm Partner Challenge: Hangman

A student starting coding a Hangman game, but lost interest. The only thing basically missing is the main function. You and your partner are going to complete that game.

Note: The student thought that initially the word to be guessed would be displayed as a series of underlines separated by spaces. For example, if the word to be guessed was *materialism* the initial display would be

 _ _ _ _ _ _ _ _ _ _ and after a number of guesses it might be
 _ A _ E _ I A _ I _ _

Let's call this word (the one guessed so far) `guess` and the secret word they are trying to determine `secret`. So `secret` might be 'MATERIALISM' and `guess` might be '_ A _ E _ I A _ I _ _'. The student thought he could test whether the user guessed the secret word by using

```
guess == secret
```

but that didn't work. When the user guessed the secret work `guess` would be 'M A T E R I A L I S M' and `word` would be 'MATERIALISM'. To fix this problem she wrote the function `addBlanks` that adds blanks between each letter. Now she could test to see if the user guessed the word by

```
guess == addBlanks(secret)
```

Partners: _____

Part 1: The task in English

We can consider many games to have three parts: the preparation part, the turn-based, looping part (including a test on whether to stop looping), and the wrap-up. For example, for a card game the preparation part might be to shuffle the cards, deal cards to each player, and place the remainder of the deck in the center. The turn-based part might be that each player in-turn picks up a card from the deck and lays out cards according to some rules. The test for stopping the loop might be when one player is out of cards. The wrap-up might be each player counting the points they have, then deciding on the winner. In psuedo-Python in might look like this:

```
def play cards:
  # preparation part
  shuffle deck
  deal out cards

  # turn based part
  while each player still has cards:
    for each player:
      take card from deck
      lay out allowed sets of cards
      discard card

  # wrap-up
  for each player:
    count points
  decide on winner
```

For the hangman game, describe in English what might be in each of the following sections:

PREPARATION:

TURN-BASED PART

WHEN TO STOP TURN-BASED PART

WRAP UP

Part 2: Pre-existing functions

The student that started the project, already created a number of functions we can use (see the code in `hangman.py`). Match (by drawing a line between the two) whether we might use the functions listed on the right in the preparation, loop-test, loop, wrap-up part of the code. **Note:** not all parts on the left will have associated functions on the right.

Preparation

`print_hangman_image(mistakes)`

`replace_letter(letter, word,
wordSoFar)`

Loop-test

`get_random_word()`

Loop

`createBlankWord(length)`

`addBlanks(word)`

Wrap-up

Part 3: Code the game

Download the files `hangman.py`, `hangman_lib.py`, and `word_list.txt` into the same directory.

Write the function `hangman`.

Extra XP for implementing any of the following:

- have the program print “You've already guessed this letter” if the user enters the same letter twice. (and the guess doesn't count)
- Right now the user needs to type in capital letters. Modify the code so that the user can type in upper or lower case letters. (if response is a string, `response.upper()` converts it to upper case).
- The less than (<) and greater than (>) work for strings. Make sure the user enters a letter as a guess and if not print 'You can only guess letters, not numbers or symbols!' Also make sure the user types in only one letter.
- Add features of your own (be sure to say what features you added in the initial comment of the program).

Extra Class Time Friday: 4-6pm

I will have dedicated office hours on Friday from 4-6 in the basement of Trinkle. Come sit at the tables in the corridor, grab some desk space in one of the labs, sit on the couch in the computer science office. I will be floating around providing assistance or come to me in B20. If people are still around at 6 I will stay an additional hour.

Submit: due Friday 6pm.

Submit the answers to parts 1 and 2 in person or in my mailbox no later than Friday 6pm. Email your code for part 3 to submit.o.bot@gmail.com (subject = 110 midterm).