EECS 198 Homework 4 (10 points)
Due: Tuesday, October 23 at midnight

Part 1: Codeacademy - Functions (0 points, but helpful for later parts of homework)

On Codeacademy, go to the *Learn Python* course (https://www.codecademy.com/learn/learn-python). For this homework assignment, please complete lesson 4, "Functions". (You can get to this lesson by going to "Syllabus" and then clicking on Lesson 4.) You only need to complete the first interactive lesson; you do not need to complete the rest of the lessons in the module (e.g., you don't need to do Taking a Vacation, Number Guess, or Functions Quiz).

Part 2: Madlibs Program (8 points)

For this part of the homework, you are going to write a Python program. Name your program **homework4.py**, and upload this file to Canvas. Your program should run using Python 2. Please make sure that your program runs with no errors before submitting it. If your program doesn't run without crashing, then you will receive 0% credit for it.

You are going to create a program that allows the user to play a Madlibs-inspired game. Madlibs is a game where you take a story, remove certain words, and then ask the user for random words (of the correct part-of-speech) to fill in the story, often with hilarious results! (You can find some examples of this game on the Madlibs website, www.madlibs.com/printables/, or on other, similar websites, such as www.madtakes.com/.)

Pick a short story from one of these websites (or write your own!). Then, ask the user to give you the words to fill in the blanks. For example, if one of the blanks in the story says "adjective", ask the user, "Please give me an adjective." Then, save the user's answers. Once the user has given you words for all of the blanks, print out the new story, with the user's words in the story instead of the blanks.

For this assignment, you **must** write a function **askUser** that takes one argument, a string indicating what word you are asking for (e.g., "adjective", "noun", "past tense verb"). This function should ask the user to input an appropriate word, and then return the inputted word from the function. For example, if you give the function "noun", the function should ask the user "Please give me a noun" and then return whatever word the user inputs.

Other than that, you can structure your program however you want! Before starting a project like this, it's helpful to sit down and plan out the steps of your program and how you want it to function. This can help you avoid getting stuck later on.

Want to try more?? (This part is optional, and you should not submit it for your graded assignment.)

- Go to www.ashley-bovan.co.uk/words/partsofspeech.html and download the lists of nouns, adjectives, adverbs, and verbs.
- Pick a story that only includes nouns, adjectives, adverbs, and verbs for the fill-in-the-blank words.
- Then randomly generate a complete story. To do this, for each blank in your story, randomly choose a word from one of the word lists that you downloaded (e.g., if the blank is a noun, randomly choose a word from the lists of nouns that you downloaded).
- HINT: You'll need to read the word lists into your Python program in order to do this. Suppose that you want to read in the "31K Verbs.txt" file. First, make sure that this file is in the same folder as your Python program. Then, you can use the following code to read in the file:

  ```
  with open("31K Verbs.txt","r") as verbsFile:
        verbs = verbsFile.readlines()
  ```

  Now, `verbs` will be a list of all of the lines in your file. Essentially, it will be a list of strings that you read in. One thing to be aware of is that each element in your list will have an extra new line, or enter, character at the end of it. So, each element will look something like `"run\n"`. This is because of how we're reading in the file. If you want to get rid of this, you can use the following line of code:

  ```
  verbs = [verb[:-1] for verb in verbs]
  ```

  This is called a list comprehension, which we haven't gotten a chance to talk about in class yet. If you're interested, there's more information about list comprehensions at www.datacamp.com/community/tutorials/python-list-comprehension. You can also find out more about file input and output at https://www.guru99.com/reading-and-writing-files-in-python.html. As always, if you have questions, feel free to ask at office hours or on Piazza!

Part 3: Complete homework survey (2 points)

Go to https://goo.gl/forms/K5DQhu8hSJ8sr5392 and complete a short survey about this homework assignment. If you fill out the required questions in the survey, you will get full points for this part (regardless of your answers to the survey questions). This survey will be helpful to us as we design future homework assignments.