The filename for your homework must be HW3_Python. Submit to your GitHub repository.

Exercises compiled by Torbjörn Lager.

1. Define a procedure histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:

    ```
    ****
    *********
    *******
    ```

2. The function max() from exercise 1) and the function max_of_three() from exercise 2) will only work for two and three numbers, respectively. But suppose we have a much larger number of numbers, or suppose we cannot tell in advance how many they are? Write a function max_in_list() that takes a list of numbers and returns the largest one.

3. Write a program that maps a list of words into a list of integers representing the lengths of the correponding words.

4. Write a function find_longest_word() that takes a list of words and returns the length of the longest one.

5. Write a function filter_long_words() that takes a list of words and an integer n and returns the list of words that are longer than n.

6. Write a version of a palindrome recognizer that also accepts phrase palindromes such as "Go hang a salami I'm a lasagna hog.", "Was it a rat I saw?", "Step on no pets", "Sit on a potato pan, Otis", "Lisa Bonet ate no basil", "Rise to vote sir", or the exclamation "Dammit, I'm mad!". Note that punctuation, capitalization, and spacing are usually ignored.

7. A pangram is a sentence that contains all the letters of the English alphabet at least once, for example: The quick brown fox jumps over the lazy dog. Your task here is to write a function to check a sentence to see if it is a pangram or not.

8. "99 Bottles of Coke" is a traditional song in the United States and Canada. It is popular to sing on long trips, as it has a very repetitive format which is easy to memorize, and can take a long time to sing. The song's simple lyrics are as follows:

   99 bottles of coke on the wall, 99 bottles of coke.
   Take one down, pass it around, 98 bottles of coke on the wall.

   The same verse is repeated, each time with one fewer bottle. The song is completed when the singer or singers reach zero.

   Your task here is write a Python program capable of generating all the verses of the song.

9. Represent a small bilingual lexicon as a Python dictionary in the following fashion {"merry":"god", "christmas":"jul", "and":"och", "happy":gott", "new":"nytt", "year":"år"} and use it to translate your Christmas cards from English into Swedish. That is, write a function translate() that takes a list of English words and returns a list of Swedish words.

10. Write a function char_freq() that takes a string and builds a frequency listing of the characters contained in it. Represent the frequency listing as a Python dictionary. Try it with something like char_freq("abbabcbdbabdbdbabababcbcbab").

11. In cryptography, a Caesar cipher is a very simple encryption techniques in which each letter in the plain text is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 3, A would be replaced by D, B would become E, and so on. The method is named after Julius Caesar, who used it to communicate with his generals. ROT-13 ("rotate by 13 places") is a widely used example of a Caesar cipher where the shift is 13. In Python, the key for ROT-13 may be represented by means of the following dictionary:

    ```
    key = {'a':'n', 'b':'o', 'c':'p', 'd':'q', 'e':'r', 'f':'s', 'g':'t', 'h':'u',
        'i':'v', 'j':'w', 'k':'x', 'l':'y', 'm':'z', 'n':'a', 'o':'b', 'p':'c',
        'q':'d', 'r':'e', 's':'f', 't':'g', 'u':'h', 'v':'i', 'w':'j', 'x':'k',
        'y':'l', 'z':'m', 'A':'N', 'B':'O', 'C':'P', 'D':'Q', 'E':'R', 'F':'S',
        'G':'T', 'H':'U', 'I':'V', 'J':'W', 'K':'X', 'L':'Y', 'M':'Z', 'N':'A',
        'O':'B', 'P':'C', 'Q':'D', 'R':'E', 'S':'F', 'T':'G', 'U':'H', 'V':'I',
        'W':'J', 'X':'K', 'Y':'L', 'Z':'M'}
    ```

    Your task in this exercise is to implement an encoder/decoder of ROT-13. Once you're done, you will be able to read the following secret message:

      Pnrfne pvcure? V zhpu cersre Pnrfne fnynq!

Note that since English has 26 characters, your ROT-13 program will be able to both encode and decode texts written in English.

12. Define a simple "spelling correction" function correct() that takes a string and sees to it that 1) two or more occurrences of the space character is compressed into one, and 2) inserts an extra space after a period if the period is directly followed by a letter. E.g. correct("This   is very funny and   cool.Indeed!") should return "This is very funny and cool. Indeed!"

13. The third person singular verb form in English is distinguished by the suffix -s, which is added to the stem of the infinitive form: run -> runs. A simple set of rules can be given as follows:

   - If the verb ends in y, remove it and add ies
   - If the verb ends in o, ch, s, sh, x or z, add es
   - By default just add s

   Your task in this exercise is to define a function make_3sg_form() which given a verb in infinitive form returns its third person singular form. Test your function with words like try, brush, run and fix. Note however that the rules must be regarded as heuristic, in the sense that you must not expect them to work for all cases. Tip: Check out the string method endswith().

14. In English, the present participle is formed by adding the suffix -ing to the infinite form: go -> going. A simple set of heuristic rules can be given as follows:

   - If the verb ends in e, drop the e and add ing (if not exception: be, see, flee, knee, etc.)
   - If the verb ends in ie, change ie to y and add ing
   - For words consisting of consonant-vowel-consonant, double the final letter before adding ing
   - By default just add ing

   Your task in this exercise is to define a function make_ing_form() which given a verb in infinitive form returns its present participle form. Test your function with words such as lie, see, move and hug. However, you must not expect such simple rules to work for all cases.