Python for Engineers 0509-1820, Fall 2020-2021

Home Assignment 2
Loops and Lists

General directions:

- Read questions thoroughly and make sure your programs fulfill the required task.
- The assignment should be solved on your own!
- Follow the submission guidelines written on the website. In particular, all questions ought be submitted in the attached template file ex2_012345678.py, after replacing the digits 012345678 with your 9-digit ID number (Teudat Zehut if you have one, otherwise it's typically a number beginning with 9).
- Submission is due by: see website.
- Self testing: to verify your programs are correct and robust, run your programs with many possible inputs – the examples from this file, as well as other inputs of your choice. All outputs must be correct and your programs must never crash.
- We often use automated testing of assignments, so your outputs must match the required format **exactly** (whitespace and capitalization matter!)
  Follow the output format provided in the examples.
- Task: in this assignment you need to add your code to the given template file.
- Do not change any variable name in the template file.

Your input will be given to you in the variables predefined in the template file, currently assigned the invalid value ???. Your code should use these values, make a computation, and provide the required output.

Your code may use additional variables, if needed.

- Do not remove comments from the template file.

General Remark

For the time being, we assume that the inputs have valid types; i.e., if we say some variable's value is a list of integers, you don't need to verify it (in Real Life™, trust no user input!). Other assumptions about the input (e.g., can lists be empty?) **cannot** be made unless explicitly mentioned in the question.

Question 1

You are given a list `lst` consisting of positive integers and a positive integer `a`.

Print the **index** in `lst` of the first element divisible by `a`. In case no such element exists, print `-1`. Recall that `a>0` but `lst` might be empty.

Example 1. For the input lst = [1, 2, 3, 4, 5], a = 3 the output is
2
Example 2. For the input lst = [1, 2, 3, 4, 5], a = 8 the output is
-1

Question 2

You are given a list `lst2` consisting of strings (elements of type `str`). Compute the average length of strings in `lst2` and print the number of strings (strictly) longer than average.

Solve the question twice: first using a `while` loop, and the second time using a `for` loop. You may assume `lst2` is not empty. Note that the average might be non-integral.

Example. For the input lst2 = ['hello', 'world', 'course', 'python', 'day'] the average length is 5 and the output is
The number of strings longer than the average is: 2
The number of strings longer than the average is: 2

(printed twice since you need to do both `while` and `for`)

Question 3

You are given a list `lst3` of numbers. Write code that prints the **sum** of **products** of consecutive pairs of numbers in `lst3`. If the list is empty, print 0; if it consists of a single number, print this number.

Example. For the input lst3 = [0, 1, 2, 3, 4] the output is
20
(since $0 \cdot 1 + 1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 = 20$)

Question 4

You are given a list `lst4` consisting of positive integers. There are at least two numbers in `lst4`. Write code that prints a new list, which is a sub-sequence of `lst4`; that is, the source of all elements of the new list are lst4, and they maintain the order from lst4, but elements of lst4 not satisfying the following condition are skipped.
The property satisfied by the new list is that the absolute difference of every consecutive pair in it is strictly greater than the absolute difference of previous consecutive pairs.

Example 1. For the input lst4 = [1, 2, 4, 6, 5, 9] the output is
[1, 2, 4, 9]

Explanation: the new list satisfies $|1-2| < |2-4| < |4-9|$. The element 6 was skipped since $|4-6| = |2-4|$, and the element 5 was skipped since $|4-5| < |2-4|$.

Example 2. For the input lst4 = [1, 2, 4, 8] the output is
[1, 2, 4, 8]

(no elements skipped since $|1-2| < |2-4| < |4-8|$)

Example 3. For the input lst4 = [1, 3, 0, 2] the output is
[1, 3, 0]

(only last element is skipped since $|3-0| > |0-2|$)

You may find the built-in function `abs` useful. Read about it in the documentation: https://docs.python.org/3/library/functions.html or just type `help(abs)` in the Python shell.

Question 5

You are given a string `my_string` and a positive integer `k`. Write code that prints the first substring of `my_string` of length `k` all of whose characters are identical (lowercase and uppercase are different).
If none such exists, print an appropriate error message (see Example 4 below). In particular, the latter holds when `my_string` is empty.

Hint: recall how to easily create a string consisting of a single character repeated `k` times.

Example 1. For the input my_string = "abaadddefggg", k = 1 the output is
For length 1, found the substring a!

Example 2. For the input my_string = "abaadddefggg", k = 3 the output is
For length 3, found the substring ddd!

Example 3. For the input my_string = "abccccccd", k = 2 the output is
For length 2, found the substring cc!

Example 4. For the input my_string = "abaadddefggg", k = 9 the output is
Didn't find a substring of length 9

Good luck!