

## Day 1: Extended Problems

Advanced basics in Python involve nested lists, more data types, and some more complex problems to solve. Here are a set of advanced problems to test your knowledge of what we covered earlier today, and to test your understanding of this afternoon's extra content!

If you're new to programming and find these challenging, *don't worry*. Some of these problems are complicated, and if you've made it this far you're doing very well indeed! Ask a tutor for help, and for praise on your progress.

### Iteration

**Problem 1.** Use a for-loop to find the length of a list.

**Problem 2.** Given a list of integers, use a for-loop to find its maximum element; then do the same to find its minimum element.

**Problem 3.** Find the average of a list of integers.

**Problem 4.** Given a list of integers, can you find the runner-up? For example, given `[2, 3, 3, 6, 4, 5]`, the maximum is 6 and the runner-up is 5.

**Problem 5.** Suppose you are given the list `[1, 1]`, if you add the last two numbers and append it to the list, the new list will be `[1, 1, 2]`. If you repeat this addition sequence on the new list three more times, you obtain `[1, 1, 2, 3, 5, 8]`. Starting with `[1, 1]`, can you print the resulting list if you carry out this addition sequence 50 times? *You've just printed the first 50 elements of the Fibonacci sequence.*

**Problem 6.** Given two lists  $L$  and  $R$ , sum their elements in pairs using a for-loop; for example, if  $L = [1, 2, 3]$  and  $R = [3, 2, 1]$ , then their sum should be `[4, 4, 4]`.

**Problem 7.** Given a square matrix of size  $N \times N$  represented as a list of list, print all the elements in its left diagonal. For example, given the  $3 \times 3$  matrix

`[[11, 2, 4], [4, 5, 6], [10, 8, -12]]`

you are expected to print `11 5 -12` (*note that the delimiter is space*). Can you find the sum of these elements?

**Problem 8.** Now print all the elements in its right diagonal. Can you find the sum of these elements?

**Problem 9.** What is the absolute difference between the sum of elements in the left diagonal and the sum of elements in the right diagonal?

**Problem 10.** Pair up corresponding entries from 2 lists to form a nested list of pairs.

**Problem 11.** Given a nested list of pairs separate the pair into individual lists.

**Problem 12.** Given a nested list flatten the list into a single list.

### *If and Conditionals*

A quick recap on `if` statements, just to reinforce the concepts!

**Problem 13.** Given a list of integers, how many of its elements are positive, negative, and are zeros?

**Problem 14.** Given a list of integers, how many of its elements are multiples of 3? For example, given the list `[5, 6, 9, 10, 15, 17, 18, 21]`, five of its elements are multiples of 3.

### *Strings*

**Problem 15.** Write out all the characters in a string one per line

**Problem 16.** Write out all the characters backwards

**Problem 17.** Write out the first and the last, second and second last...

**Problem 18.** Write a program which checks if a string is palindromic.

### *Advanced string questions*

**Problem 19.** You are given an un-ordered array consisting of integers without any duplicates. You are allowed to swap any two elements. You need to find the minimum number of swaps required to sort the array in ascending order

**Problem 20.** Two strings are anagrams of each other if the letters of one string can be rearranged to form the other string.

Given two strings, *a* and *b*, that may or may not be of the same length, determine the minimum number of character deletions required to make *a* and *b* anagrams. Any characters can be deleted from either of the strings.

**Problem 21.** Two strings are anagrams of each other if the letters of one string can be rearranged to form the other string. Given a string, find the number of pairs of sub-strings of the string that are anagrams of each other.

**Problem 22.** You are given 3 strings:  $f$ ,  $s$ , and  $t$ .  $t$  is said to be a shuffle of  $f$  and  $s$  if it can be formed by interleaving the characters of  $f$  and  $s$  in a way that maintains the left to right ordering of the characters from each string.

For example, given  $f = \text{"abc"}$  and  $s = \text{"def"}$ ,  $t = \text{"dabecf"}$  is a valid shuffle since it preserves the character ordering of  $f$  and  $s$ . So, given these 3 strings write a program that detects whether the third string is a valid shuffle of first and second.