Department of Computer Science and Engineering
PES University, Bangalore, India

# Lecture Notes
# Python for Computational Problem Solving
# UE23CS151A

## Lecture #95

*Problem Solving – Functional Programming*

By,
Prof. Sindhu R Pai,
Anchor, PCPS - 2023
Assistant Professor
Dept. of CSE, PESU
&
Prof. Srividhya Lakshmi R
Assistant Professor
Dept. of CSE, PESU

Many Thanks to
Dr. Shylaja S S (Director, CCBD and CDSAML Research Centers, Former Chairperson, CSE, PES University)
Prof. Chitra G M(Asst. Prof, Dept. of CSE, PCPS Anchor – 2022)

# Practice Programs

**Try solving these problems using functional programming construct such as map, filter, reduce, zip, min, max, lambda and list comprehension. Solutions for a few are available in this link: https://drive.google.com/file/d/1ewdoo9ekjExSc6OQsPDfzIwGhdDETSE0/view?usp=drive_link**

1. Given a list having numbers in terms of strings, nums_strs = ["4", "8", "6", "5", "3", "2", "8", "9", "2", "5"], create a new list having the integers.

2. Given a list of numbers, numbers = [-2, -1, 0, 1, 2]. Create the list with its absolute values

3. Given a list of numbers, numbers = [-2, -1, 0, 1, 2]. Create a list with its absolute numbers if the number is negative.

4. Given a list of numbers, numbers = [9, 12, 4, 2, 7]. Create a new list by adding 1 to it if the number is even and subtract 1 from it if the number is odd.

5. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]. Create a new tuple by capitalizing the words in the list.

6. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]
Create a new tuple by capitalizing the words in the list only if it is in upper case.

7. with_dots = ["PYTHON..", ".computational..", "..PROBLEM", "solving..", "for", "first", "ye..ar."]. Create a list by removing leading and trailing dots.

8. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]

Create a list having the ordinal values of all the first letters of words in a given list

9. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]

Create a list having the tuple for every string in a given list. The tuple contains all the letters in that string separately.

10.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: [9,14,18]

11.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: [[3,1,5],[5,2,7],[6,3,9]]

12.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: [3,1,5,5,2,7,6,3,9]

13.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: [3,5,6,1,2,3,4,5,7,9]

14.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: [14,10,21]

15:

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: {3:(5,6),1:(2,3,4),5:(7,9)}

16.

m1 = (3,5,6)

m2 = (1,2,3,4)

m3 = (5,7,9)

Expected output: {3:11,1:9,5:16}

17. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]

Expected output: [{'PYTHON': 6}, {'computational': 13}, {'PROBLEM': 7}, {'solving': 7}]

18. List all the files and directories in the current folder along with the size of the file.

19. strings_li = ["PYTHON", "computational", "PROBLEM", "solving"]

Get the unique letters from each word and make it as a string. order of letters can be ignored.

Expected output: ['NHYPTO', 'olatnumcip', 'BEPLMRO', 'solnvgi']

20. Find the sum of digits of a number entered by the user.

21. Merge two lists such that new list contains one from first and the other from second. Once the list is complete, append all from the longer list to the new list.

Example: m = [23,45,67]    n = [12,65,98,23,55]

output_list = [23,12,45,65,67,98,23,55]

22. Three lists given. Output must be sum of all numbers in that corresponding index

l1 = [11,22,14]

l2 = [45,77,88]

l3 = [90,99,55,10]

output_list = [sum1,sum2,sum3]

23. Three lists given. New list should have the remainder of every element divided by 10 in that index

l1 = [11,22,14]

l2 = [45,77,88]

l3 = [90,99,55,10]

output_list = [[1,5,0],[2,7,9],[4,8,5]]

24. Three lists given. New list should have the first digit of every element from the given list.

l1 = [11,22,14]

l2 = [45,77,88]

l3 = [90,99,55,10]

output_list = [[1,4,9],[2,7,9],[1,8,5],[None, None,1]]

**-END-**