Lab Assignment 2 (Practice Lab)-Even 2021

OPEN SOURCE SOFTWARE LAB (15B17CI575)

Important Instructions:

1. The course of Open Source Software Lab (OSS Lab) is of 1 credit where 1 lab per week is to be conducted.
2. All students are required to attend at least 80% labs.
3. All students are required to maintain the lab records which are mandatory to be submitted.

Topic Coverage: Python

1. Create a python script to print hello, world! four times.
2. Write a function “duplicate” to find all duplicates in the list.
3. Write a function group (list, size) that take a list and splits into smaller lists of given size.
4. Write a function “lensort” to sort a list of strings based on length.
5. Write a function extsort to sort a list of files based on extension.
6. Use Python Built-in Functions ‘open’, ‘read’, ‘’readline, ‘write’,’writeline’ to work with files.
7. Compute the number of characters, words and lines in a file.
8. Write a program reverse.py to print lines of a file in reverse order.
9. Write a program to print each line of a file in reverse order.
10. Write a program wrap.py that takes filename and width as aruguments and wraps the lines longer than width.
11. Python provides a built-in function map that applies a function to each element of a list. Provide an implementation for map using list comprehensions.
12. Python provides a built-in function filter(f, a) that returns items of the list a for which f(item) returns true. Provide an implementation for filter using list comprehensions
13. Write function triplets that takes a number n as argument and returns a list of triplets such that sum of first two elements of the triplet equals the third element using numbers below n. Please note that (a,b, c) and (b, a, c) represent same triplet.
14. Write a python function parse_csv to parse csv (comma separated values) files.
15. Generalize the above implementation of csv parser to support any delimiter and comments.
16. Generalize the above implementation of csv parser to support any delimiter and comments.
17. Write a function mutate to compute all words generated by a single mutation on a given word. A mutation is defined as inserting a character, deleting a character, replacing a character, or swapping 2 consecutive characters in a string. For simplicity consider only letters from a to z.
18. Write a function nearly_equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
19. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?
20. Write a program to find anagrams in a given list of words. Two words are called anagrams if one word can be formed by rearranging letters of another. For example ‘eat’, ‘ate’ and ‘tea’ are anagrams.