2810ICT – Workshop 5, 2019

Cocalc Editor, <https://cocalc.com/projects/d73e187a-abf4-4e32-a38d-05759cc7020a/files/Workshop%205.term?session=default>

**Program 1:**

import string

punctuations = '''!()-[]{};:'"\,<>./?@#$%^&\*\_~'''

data = input("Enter a string: ")

# Translates and Replaces Chars with Desired Charecters

data = data.translate(str.maketrans('', '', string.punctuation))

# Changes the Input to Uppercase and splits it into a list

data\_arr = data.upper().split(' ')

a = {}

# Checks if a Word Appears More than Once in the list

# It will Append it to Add +1

for i in data\_arr:

if i not in a:

a[i] = 1

else:

a[i] = a[i] + 1

# Sorts all the Items, key=lambda a: a[0] to Sort Alphabetically

data = sorted(a.items())

# Removes Tuple Brackets

data = [', '.join(map(str, x)) for x in data]

# Prints Vertically

print(\*data, sep = '\n')

**Program 2:**

import math

# Function to find the word counts

def count(str):

# Create an empty dictionary

counts = dict()

# Split the line into words

words = str.split()

# Iterate over each word in line

for word in words:

# make it uppercase

word = word.upper()

# Check if the word is already in dictionary

if word in counts:

# Increment count of word by 1

counts[word] = counts[word] + 1

else:

# Add the word to dictionary with count 1

counts[word] = 1

# Sorting on the basis of length of words

counts = dict(sorted(counts.items(), reverse=True))

# Sorting on the basis of count for words

counts = dict(sorted(counts.items(), key=lambda x: x[1], reverse=True))

# Return the dictionary

return counts

# Function to print the histograms using dictionary

def printHist(counts):

# Get the total number of words

# Extract list of counts and then sum it up

total = sum(list(counts.values()))

# Print the histogram from counts

for word in list(counts.keys()):

# Find the percentage for the word

percent = math.floor(counts[word]\*100/total)

# Find the number of stars to be printed

stars = math.ceil(percent/5)

print(f"{word} : [{'\*'\*stars}] {percent}%")

if \_\_name\_\_ == "\_\_main\_\_":

# Take the input

text\_file = open("test.txt", "r")

#read whole file to a string

str = text\_file.read()

# Get the counts

counts = count(str)

# Print the histogram

print()

printHist(counts)

print()

**Program 5A:**

import re

# Reads the File

with open('2375633.txt') as f:

contents = f.read()

# Reviews the File for all data including an @ & .org

contents = re.findall('\S+@\S+.org\S+', contents)

# Removes Duplicates

contents = list(dict.fromkeys(contents))

print(contents)

**Program 5B: (INCORRECT, BUT CLOSE)**

from datetime import datetime as dt

import re

start = dt.strptime('15:00:00' , '%H:%M:%S')

end = dt.strptime('19:00:00' , '%H:%M:%S')

s = open('test.txt', 'r' , encoding ='utf-8')

contents = s.read()

lst = re.findall('[^,;\s]+.edu[^,;\s]+',contents)

with open('test.txt', 'r') as f:

for line in f:

ts= dt.strptime(line.split()[0],'%H:%M:%S')

if ts>start and ts<end:

for j in range(len(lst)):

if lst[j] in line:

print(line)