**Aim:**

**LAB # 9**

**(TEXT FILE)**

**EXERCISE – 1**

Write a Program to READ AND WRITE SOME TEXT FROM notepad File**.**

**OR**

Write a program to count frequency of characters in a given TEXT file**. (OPTIONAL QUESTION)**

## Description:

Traverse each character in the file and its occurrence is stored in the dictionary. Here, We are using the concept of files and a Dictionary.

## Files:

Python provides basic functions and methods necessary to manipulate files by default. Some of the useful functions in this program are:

## open():

Before we can read or write a file, we have to open it using Python‟s built-in open(). This function creates a file object, which would be utilized to call other support methods associated with it.

## read():

The **Syntax** for opening a file object in Python is:

file\_object = open(“Name of the file”,”Mode of the file”)

If we need to extract string that contains all characters in the file, We can use read(). The

**Syntax** for read() in Python is:

str = file\_object.read()

## Dictionary:

The dictionary is Python‟s built-in mapping type. Dictionaries map keys to values and these

<key,value> pairs provides a useful way to store data in python. The only way to access the value part of the dictionary is by using the key.

We can specify the dictionary <key,value> pairs between „{„ and „}‟. <key,value> pairs are specified as a list(separated by commas).

## Algorithm:

**Input:** A File

**Output**: A Dictionary and A Message Step1: Start

Step2: Read file\_name

Step3: Open the specified file\_name in read mode

Step4: Read all characters of the file into the string file\_contents Step5: Create an empty dictionary Dict

Step6: Find the name of the file and store it in file\_name

Step7: Repeat Steps 8 to 10 until the end of file\_contents is reached

Step8: Check whether the ith character is present in the dictionary Dict as a key element or not. If yes, goto Step9. Otherwise, goto Step10

Step9: Increment the value of ith key element by 1 and continue with the next iteration Step10: Assign the value of ith key element to 1

Step11: Display keys and corresponding values in the dictionary Dict

Step12: Check whether the file\_name ends with .c or not. If yes, goto Step13. Otherwise, goto Step14

Step13: Display 'Input file is C program file' and goto Step18

Step14: Check whether the file\_name ends with .py or not. If yes, goto Step15. Otherwise, goto Step16

Step15: Display 'Input file is Python program file' and goto Step18

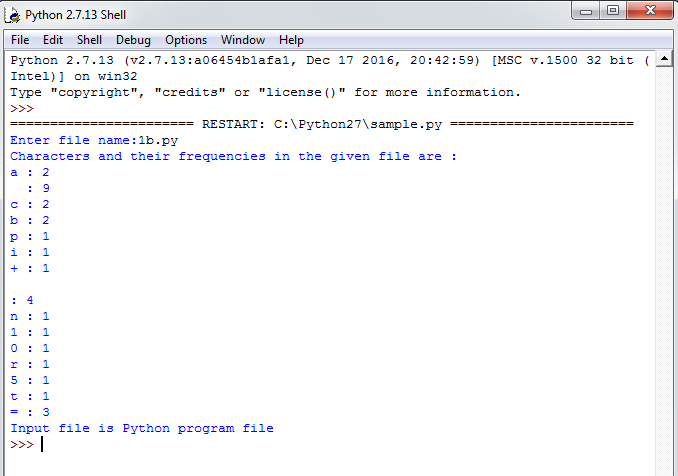
Step16: Check whether the file\_name ends with .txt or not. If yes, goto Step17. Otherwise, goto Step18

Step17: Display 'Input file is Text file' and goto Step18 Step18: Close the file

Step19: Stop

## Sample Input:

**Output:**



.

## Questions:

1. How to initialize a dictionary in python?
2. How to get all the keys from the dictionary?
3. When to choose dictionary instead of a list?
4. Write a function to sort a list?
5. Explain Python's zip() function?
6. Write a function to compare two lists?
7. What is the difference between r+ and w+ modes?
8. What is the purpose of tell()?
9. Which function is used to read single line from file?
10. What is the output of the following snippet of code? total={}

def insert(items):

if items in total:

total[items] += 1

else:

insert('Apple') insert('Ball') insert('Apple')

print (len(total))

total[items] = 1



## Aim:

**EXERCISE - 2**

Write a program to print each line of a file in reverse order.

## Description:

Traverse each line of the file and we have to display every line contents in reverse order.

Here, We are using the some of the basic methods of files and a string method strip().

Python provides basic functions and methods necessary to manipulate files by default. We can do most of the file manipulations using a file object.

## open():

Before we can read or write a file, we have to open it using Python‟s built-in open(). This function creates a file object, which would be utilized to call other support methods associated with it.

The **Syntax** for opening a file object in Python is:

file\_object = open(“Name of the file”,”Mode of the file”)

## close():

This method of a file object flushes any unwritten information and closes the file object.

Python automatically closes a file when the reference object of a file is reassigned to another file.

The **Syntax** for closing a file object in Python is: file\_object.close()

## readlines():

This method will returns every line of the file as a list. The **Syntax** for readlines() in Python

is:

## strip():

file\_lines = file\_object.readlines()

This method returns a copy of the string in which all chars have been stripped from the

beginning and end of the string (default whitespace characters).

The **Syntax** for strip() is Python is:

str.strip([chars])



## Algorithm:

**Input:** A file

**Output**: Each line of the file in reverse order Step1: Start

Step2: Read file\_name

Step3: Open the specified file\_name in read mode Step4: Read all lines of the file into the list fileContents

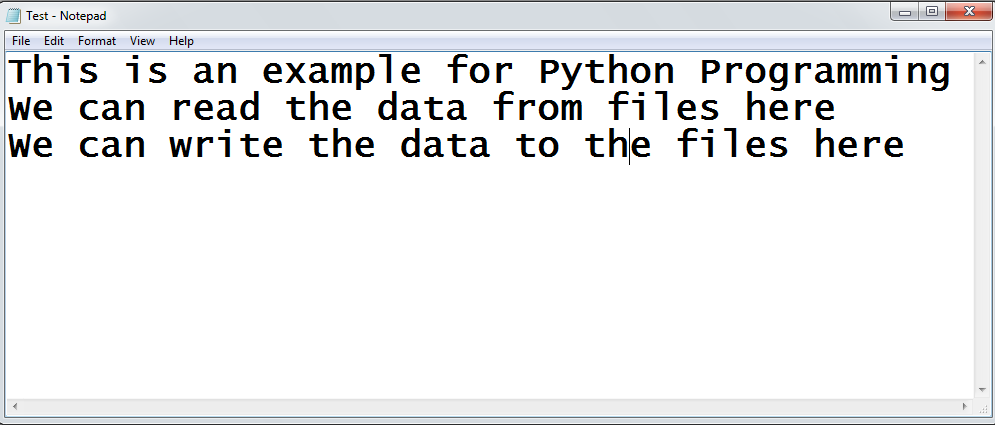
Step5: Repeat Steps 6 to 8 until end of the list fileContents is reached Step6: Create an empty string line1

Step7: Remove „\n‟ from end of the ith element in the list

Step8: Reverse the ith element in the list and display it on the screen Step9: Close the file

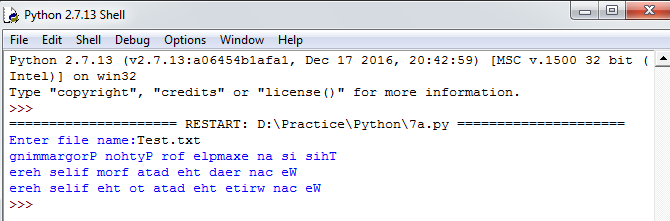
Step10: Stop

## Sample Input:





**Output:**



**Aim:**

**EXERCISE - 3**

Write a program to compute the number of characters, words and lines in a file.

## Description:

Traverse each character in the file so that we can find number of characters, words and lines in a text file. Here, We are using the some of the basic methods of files.

We already discussed about open() and close() in the previous programs.

## read():

If we need to extract string that contains all characters in the file, We can use read(). The

**Syntax** for read() in Python is:

str = file\_object.read()

## Algorithm:

**Input:** A file

**Output**: Number of characters, words and lines in an input file Step1: Start

Step2: Read file\_name

Step3: Read all characters into the string fileContents Step4: Initialize noc to 0, nol to 1 and now to 1

Step5: Repeat Steps 6 to 8 until end of the string fileContents is reached Step6: Increment noc by 1

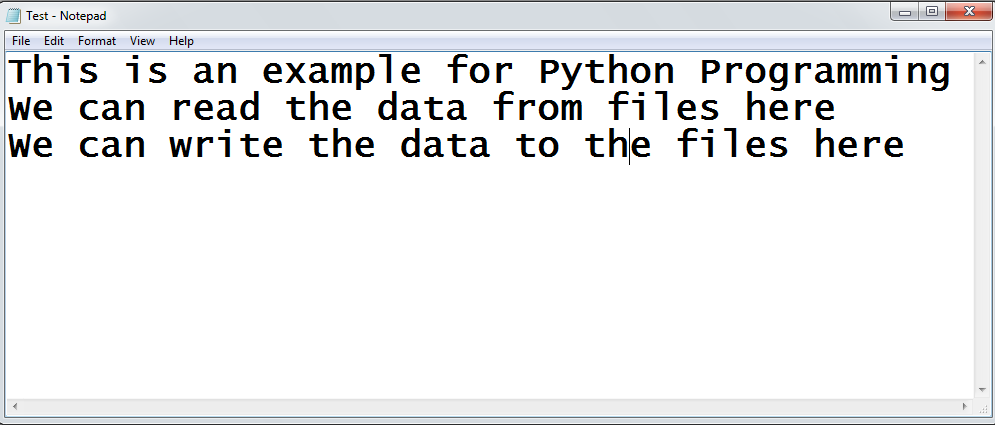
Step7: Check whether ith character in fileContents is „\n‟ or not. If yes, Increment nol by 1 Step8: Check whether ith character is („ „ or „\n‟) or not. If yes, Increment now by 1

Step9: Display nol, now and noc Step10: Close the file

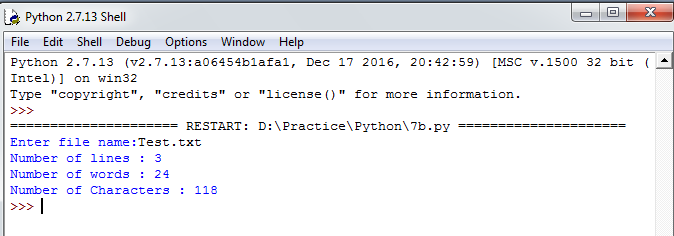
Step11: Stop



## Sample Input:



**Output:**





## Questions:

1. What is file?
2. Explain all the file processing modes supported by Python?
3. What is text file and binary file?
4. What are different file object attributes in python?
5. How to write text in to a file?
6. How to read text from the file?
7. How to get the position of the file pointer?
8. How to rename existing file?
9. How to delete file from directory?
10. Explain how can you make a Python Script executable on Unix?