

## Experiment 3.3

**Student Name:**Rajiv Paul

**UID:**20BCS1812

**Branch:** CSE

**Section/Group:**607A

**Semester:** 4th

**Date of Performance:** 28/04/2022

**Subject Name:**Programming in Python Lab

**Subject Code:** 22E-20CSP-259

### 1) Aim/Overview of the practical:

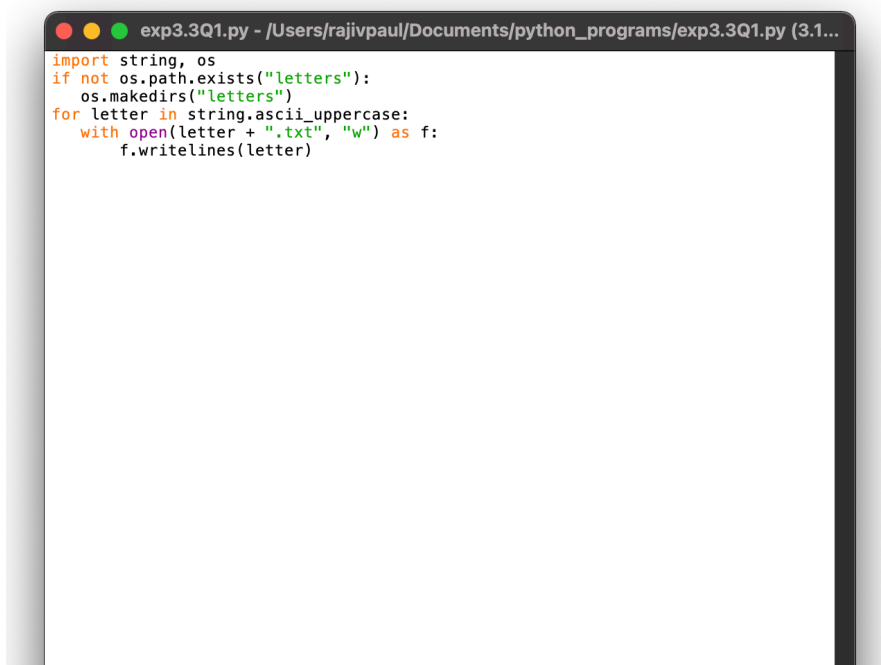
**Q1. Write a Python program to generate 26 text files named A.txt, B.txt, and so on up to Z.txt**

### 2) Task to be done/ Which logistics used:

**To write a python program to generate 26 text files named A.txt, B.txt, and so on up to Z.txt**

### 3) Algorithm/Flowchart (For programming based labs):

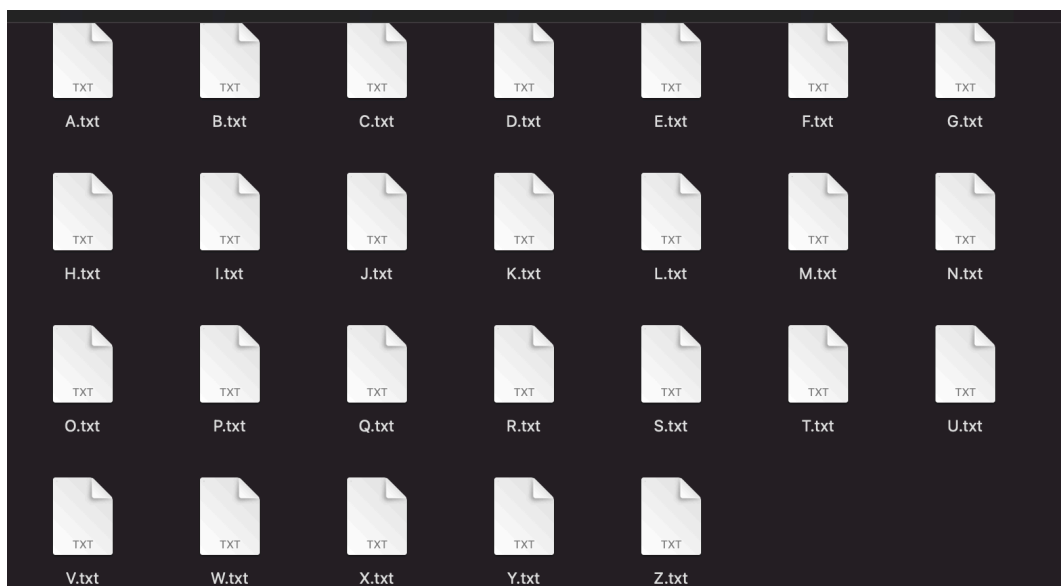
## 4) Steps for experiment/practical/Code:



```
exp3.3Q1.py - /Users/rajivpaul/Documents/python_programs/exp3.3Q1.py (3.1...  
import string, os  
if not os.path.exists("letters"):  
    os.makedirs("letters")  
for letter in string.ascii_uppercase:  
    with open(letter + ".txt", "w") as f:  
        f.writelines(letter)
```

## 5. Observations/Discussions/ Complexity Analysis:

## 6. Result/Output/Writing Summary:



**1) Aim/Overview of the practical:**

**Q2. Write a Python program to create a file where all letters of English alphabet are listed by specified number of letters on each line**

**2) Task to be done/ Which logistics used:**

**To write a python program to create a file where all letters of English alphabet are listed by specified number of letters on each line**

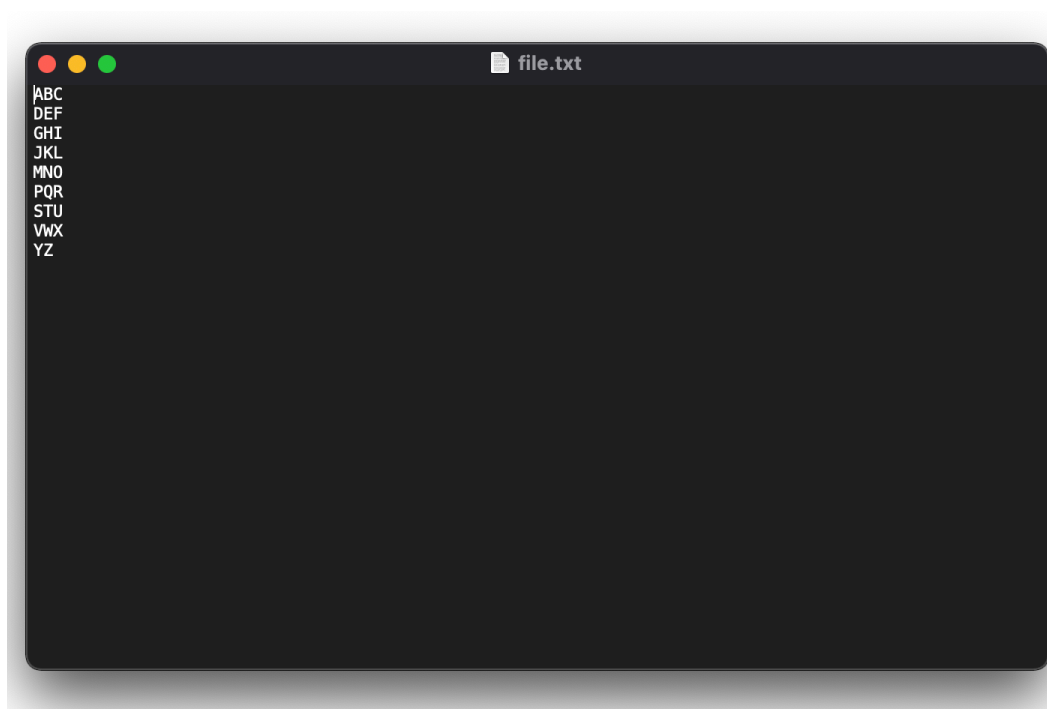
**3) Algorithm/Flowchart (For programming based labs):**

**4) Steps for experiment/practical/Code:**

```
exp3.3Q2.py - /Users/rajivpaul/Documents/python_programs/exp3.3Q2.py (3....
import string
def letters_file_line(n):
    with open("file.txt", "w") as f:
        alphabet = string.ascii_uppercase
        letters = [alphabet[i:i + n] + "\n" for i in range(0, len(alphabet), n)]
        f.writelines(letters)
letters_file_line(3)
```

## 5. Observations/Discussions/ Complexity Analysis:

## 6. Result/Output/Writing Summary:



**1) Aim/Overview of the practical:**

**Q3. Write a Python program to read a random line from a file.**

**2) Task to be done/ Which logistics used:**

**To write program to read a random line from a file.**

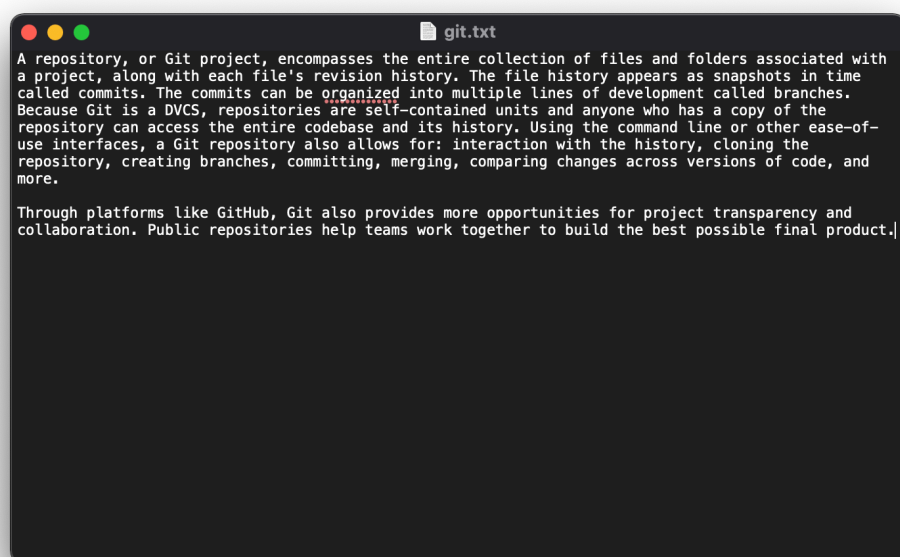
**3) Algorithm/Flowchart (For programming based labs):**

**4) Steps for experiment/practical/Code:**

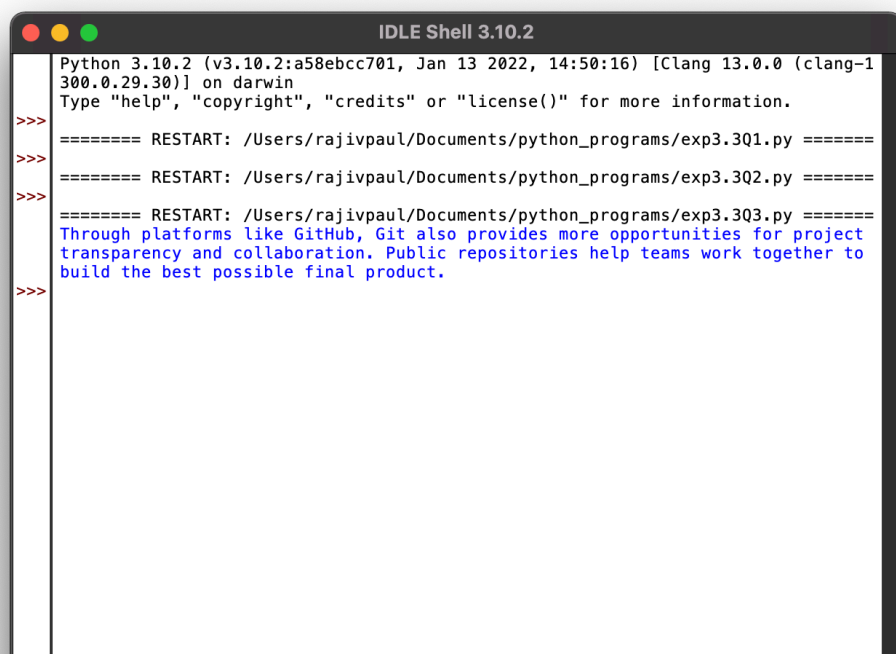
```
exp3.3Q3.py - /Users/rajivpaul/Documents/python_programs/exp3.3Q3.py (3....
import random
def random_line(fname):
    lines = open(fname).read().splitlines()
    return random.choice(lines)
print(random_line('git.txt'))
```

## 5. Observations/Discussions/ Complexity Analysis:

### git.txt file used for the program:



## 6. Result/Output/Writing Summary:



```
Python 3.10.2 (v3.10.2:a58ebcc701, Jan 13 2022, 14:50:16) [Clang 13.0.0 (clang-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/rajivpaul/Documents/python_programs/exp3.3Q1.py =====
>>>
===== RESTART: /Users/rajivpaul/Documents/python_programs/exp3.3Q2.py =====
>>>
===== RESTART: /Users/rajivpaul/Documents/python_programs/exp3.3Q3.py =====
Through platforms like GitHub, Git also provides more opportunities for project
transparency and collaboration. Public repositories help teams work together to
build the best possible final product.
>>>
```



**1) Aim/Overview of the practical:**

**Q4. Write a Python program to count the frequency of words in a file.**

**2) Task to be done/ Which logistics used:**

To write a python program to count the frequency of words in a file.

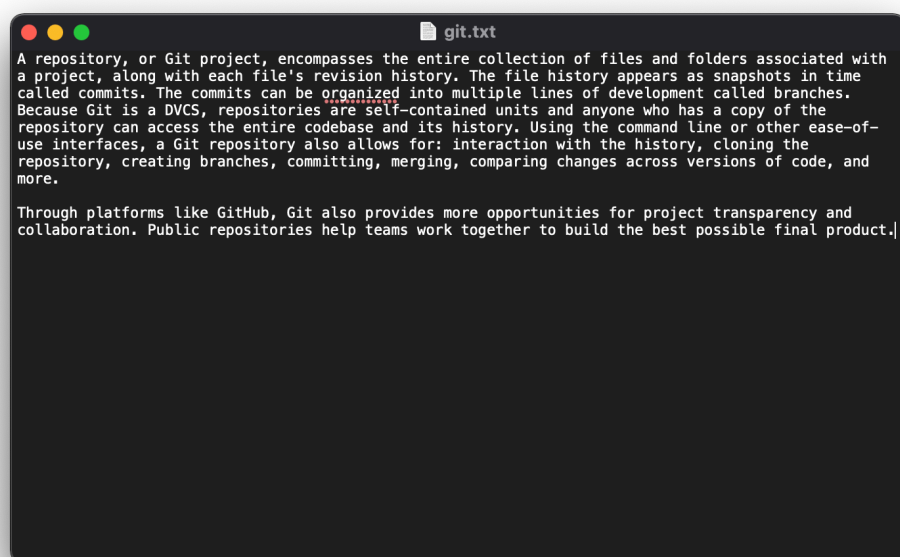
**3) Algorithm/Flowchart (For programming based labs):**

**4) Steps for experiment/practical/Code:**

```
exp3.3Q4.py - /Users/rajivpaul/Documents/python_programs/exp3.3Q4.py (3....  
from collections import Counter  
def wordcount(fname):  
    with open(fname) as f:  
        return Counter(f.read().split())  
  
print("Number of words in the file :\n",wordcount("git.txt"))
```

## 5. Observations/Discussions/ Complexity Analysis:

### git.txt file used for the program:



## 6. Result/Output/Writing Summary:

```
Python 3.10.2 (v3.10.2:a58ebcc701, Jan 13 2022, 14:50:16) [Clang 13.0.0 (clang-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.

>>>
===== RESTART: /Users/rajivpaul/Documents/python_programs/exp3.3Q4.py =====
Number of words in the file :
Counter({'the': 7, 'and': 5, 'Git': 4, 'of': 4, 'a': 4, 'with': 3, 'repository',
': 2, 'or': 2, 'project': 2, 'entire': 2, 'history.': 2, 'The': 2, 'called': 2,
'can': 2, 'repositories': 2, 'repository': 2, 'also': 2, 'A': 1, 'encompasses':
1, 'collection': 1, 'files': 1, 'folders': 1, 'associated': 1, 'along': 1, 'eac
h': 1, "file's": 1, 'revision': 1, 'file': 1, 'history': 1, 'appears': 1, 'as':
1, 'snapshots': 1, 'in': 1, 'time': 1, 'commits.': 1, 'commits': 1, 'be': 1, 'or
ganized': 1, 'into': 1, 'multiple': 1, 'lines': 1, 'development': 1, 'branches.'
: 1, 'Because': 1, 'is': 1, 'DVCS': 1, 'are': 1, 'self-contained': 1, 'units':
1, 'anyone': 1, 'who': 1, 'has': 1, 'copy': 1, 'access': 1, 'codebase': 1, 'its'
: 1, 'Using': 1, 'command': 1, 'line': 1, 'other': 1, 'ease-of-use': 1, 'interfa
ces': 1, 'allows': 1, 'for': 1, 'interaction': 1, 'history': 1, 'cloning': 1,
'creating': 1, 'branches': 1, 'committing': 1, 'merging': 1, 'comparing': 1,
'changes': 1, 'across': 1, 'versions': 1, 'code': 1, 'more.': 1, 'Through': 1,
'platforms': 1, 'like': 1, 'GitHub': 1, 'provides': 1, 'more': 1, 'opportuniti
es': 1, 'for': 1, 'project': 1, 'transparency': 1, 'collaboration.': 1, 'Public'
: 1, 'help': 1, 'teams': 1, 'work': 1, 'together': 1, 'to': 1, 'build': 1, 'best
': 1, 'possible': 1, 'final': 1, 'product.': 1})

>>>
```

### 1) Aim/Overview of the practical:

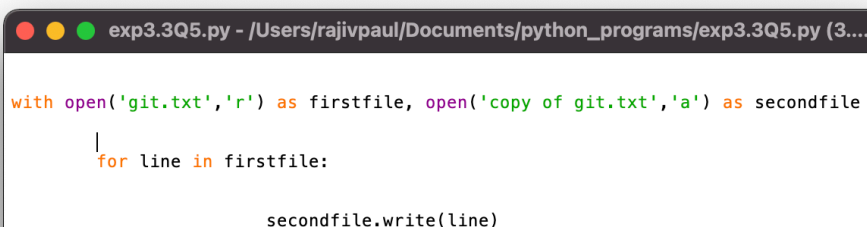
**Q5. Write a Python program to copy the contents of a file to another file.**

### 2) Task to be done/ Which logistics used:

**To write a python program to copy the contents of a file to another file.**

### 3) Algorithm/Flowchart (For programming based labs):

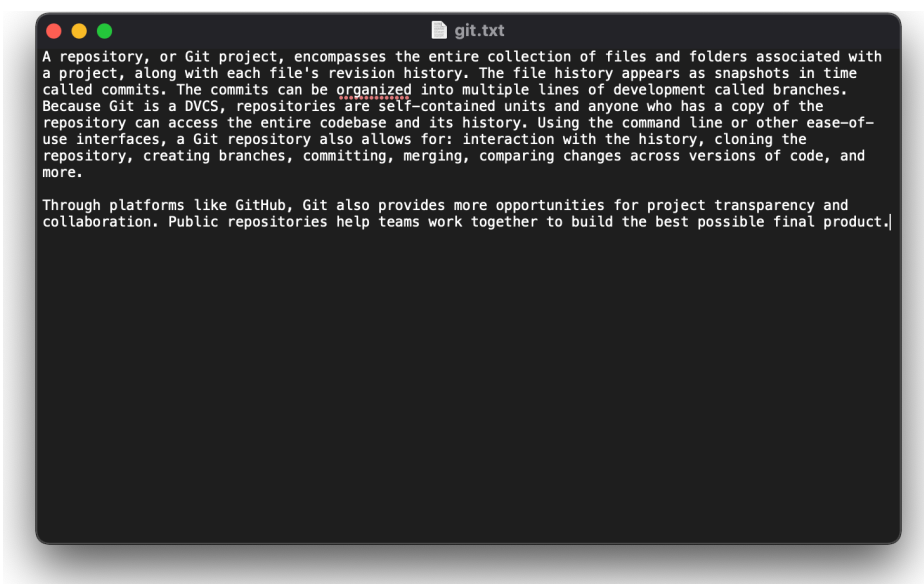
### 4) Steps for experiment/practical/Code:



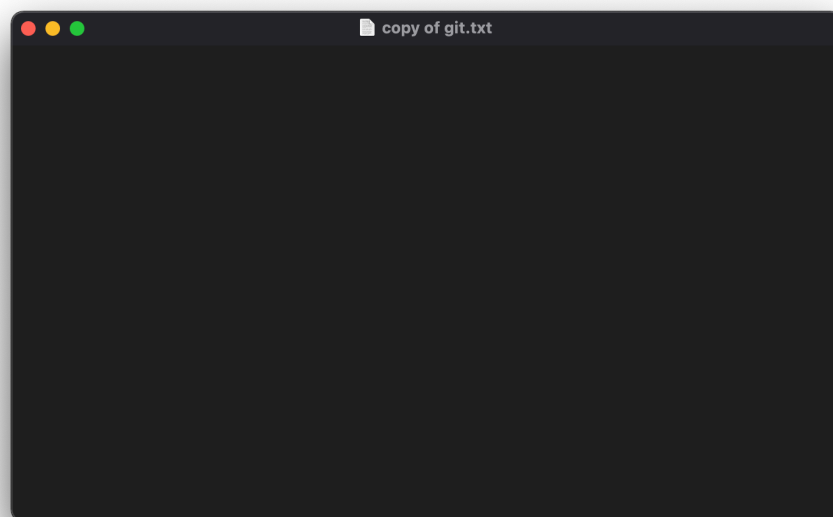
```
exp3.3Q5.py - /Users/rajivpaul/Documents/python_programs/exp3.3Q5.py (3...  
  
with open('git.txt','r') as firstfile, open('copy of git.txt','a') as secondfile  
|  
    for line in firstfile:  
  
        secondfile.write(line)
```

## 5. Observations/Discussions/ Complexity Analysis:

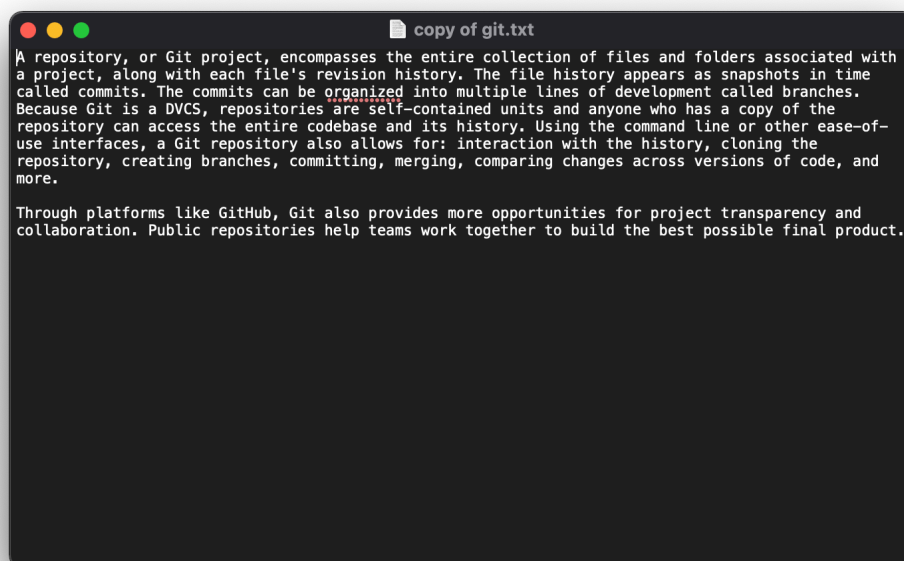
### git.txt file used for the program:



### copy of git.txt file used for the program:



## 6. Result/Output/Writing Summary:



**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			