Kalyani Government Engineering College

Department of Computer Application
Python Programming Lab – MCAN191
Year: 2023-2024
Semester: 1st Semester
Assignment 10
Name: Madhusudan Chand **Roll:** 10271023013

---

story.txt

```
There is a playground.
An aeroplane is in the sky.
The sky is pink.
Alphabets and numbers are allowed in the password.
```

poem.txt

```
I want to die as the day declines,
at high sea and facing the sky,
while agony seems like a dream
and my soul like a bird that can fly.

To hear not, at this last moment,
once alone with sky and sea,
any more voices nor weeping prayers
than the majestic beating of the waves.

To die when the sad light retires
its golden network from the green waves
to be like the sun that slowly expires;
something very luminous that fades.

To die, and die young, before
fleeting time removes the gentle crown,
while life still says: "I'm yours"
though we know with our hearts that she lies.
```

article.txt

```
This is used with singular or uncountable nouns (i.e. this egg or this music).
These refers to plural nouns (i.e. these cookies).
When the noun is omitted after this and these, they become pronouns
(i.e. turn this off when you leave).
```

matter.txt

```
THE WORLD IS ROUND
```

word.txt

```
WELL, THJS JS A WORD BY JTSELF. YOU COULD STRETCH THJS TO BE A SENTENCE
```

10.1 Write a function in python to read the content from a text file "poem.txt" line by line and display the same on screen.

Code

```
f=open("poem.txt","r")
i=f.read()
print(i)
f.close()
```

Output

```
I want to die as the day declines,
at high sea and facing the sky,
while agony seems like a dream
and my soul like a bird that can fly.

To hear not, at this last moment,
once alone with sky and sea,
any more voices nor weeping prayers
than the majestic beating of the waves.

To die when the sad light retires
its golden network from the green waves
to be like the sun that slowly expires;
something very luminous that fades.

To die, and die young, before
fleeting time removes the gentle crown,
while life still says: "I'm yours"
though we know with our hearts that she lies.
```

10.2 Write a function in python to count the number of lines from a text file "story.txt" which is not starting with an alphabet "T".

Code

```
f=open("story.txt","r")
count=0
for i in f:
    if i.split()[0][0] !='T':
        count+=1

print("Number of lines not starting with \"T\"",count)
f.close()
```

Output

```
Number of lines not starting with "T" 2
```

10.3 Write a function in Python to count and display the total number of words in a text file.

Code

```
f=open("story.txt","r")
i=f.read().split()
print("Total Number of words - ",len(i))
f.close()
```

Output

```
Total Number of words -  22
```

10.4 Write a function in Python to read lines from a text file ”notes.txt”. Your function should find and display the occurrence of the word ”the”.

Code

```
f=open("poem.txt","r")
count=0
for i in f:
    if "the" in i.split():
        count+=1
print("The occurrence of the word \"the\"",count)
f.close()
```

Output

```
The occurrence of the word "the" 7
```

10.5 Write a function display_words() in python to read lines from a text file ”story.txt”, and display those words, which are less than 4 characters.

Code

```
def display_words(f):
    m=f.read()
    m1=m.split()
    for j in m1:
        if len(j)<4:
            print(j,end=" ")

f=open("story.txt","r")
display_words(f)
f.close()
```

Output

```
is a An is in the The sky is and are in the
```

10.6 Write a function in Python to count the words ”this” and ”these” present in a text file ”article.txt”.

Code

```
import re
f=open("article.txt","r")
count=0
reg="[T|t]hese|[T|t]his"
i=f.read()
make=re.findall(reg,i)
count+=len(make)
print("Number of \"this\" and \"these\"",count)
```

Output

```
Number of "this" and "these" 8
```

10.7 Write a function in Python to count words in a text file those are ending with alphabet ”e”.

Code

```
import re
f=open("story.txt","r")

reg="(e)[^a-zA-Z]"
count=0
i=f.read()
make=re.findall(reg,i)
count+=len(make)

f.close()
print(count,"Number of words in a \"story.txt\" those are ending with alphabet \"e\"")
```

Output

```
6 Number of words in a "story.txt" those are ending with alphabet "e"
```

10.8 Write a function in Python to count uppercase character in a text file.

Code

```
import re
f=open("story.txt")
reg="[A-Z]"
count=0
for i in f:
    make=re.findall(reg,i)
    count+=len(make)

f.close()
print("Number of uppercase character",count)
```

Output

```
Number of uppercase character 4
```

10.9 A text file named "matter.txt" contains some text, which needs to be displayed such that every next character is separated by a symbol "#". Write a function definition forhash_display() in Python that would display the entire content of the file matter.txt in the desired format.

Code

```
def hash_display(f):
    data=f.read()
    print(data)
    for i in data:
        if i.isalpha():
            print(i,end="#")

f=open("matter.txt","r")

hash_display(f)
f.close()
```

Output

```
T#H#E#W#O#R#L#D#I#S#R#O#U#N#D#
```

10.10 Aditi has used a text editing software to type some text. After saving the article as WORDS.TXT, she realised that she has wrongly typed alphabet J in place of alphabet I everywhere in the article.
Write a function definition for JTOI() in Python that would display the corrected version of entire content of the file WORDS.TXT with all the alphabets "J" to be displayed as an alphabet "I" on screen.

Code

```
def JTOI(f):
    d=f.read()

    for i in d:
        if i=="J":
            print("I",end="")
        else:
            print(i,end="")
f=open("words.txt","r")
JTOI(f)
f.close()
```

Output

```
WELL, THIS IS A WORD BY ITSELF. YOU COULD STRETCH THIS TO BE A SENTENCE
```

10.11 Write a function AMCount() in Python, which should read each character of a text file STORY.TXT, should count and display the occurance of alphabets A and M (including small cases a and m too).

Code

```
def AMCount(F):
    i=f.read()
    count1=count2=0
    for j in i:
        if j is "a" or j is "A":
            count1+=1
        elif j is "m" or j is "M":
            count2+=1
    print("A or a:",count1,"\nM or m:",count2)

f=open("story.txt","r")
```

```
AMCount(f)
f.close()
```

Output

```
A or a: 11
M or m: 1
```

10.12 A binary file "Book.dat" has structure

$$BookNo, Book\_Name, Author, Price$$

. i. Write a user defined function createFile() to input data for a record and add to Book.dat. ii. Write a function countRec(Author) in Python which accepts the Author name as parameter and count and return number of books by the given Author are stored in the binary file "Book.dat".

Code

```
import pickle

def createFile():
    file = open("book.dat","ab")
    BookNo = int(input("Enter book number: "))
    Book_Name = input("Enter book Name: ")
    Author =input("Enter author: ")
    Price = float(input("Enter price: "))
    record = [BookNo, Book_Name, Author, Price]
    pickle.dump(record, file)
    file.close()

def countRec(Author):
    file = open("book.dat","rb")
    count = 0
    try:
        while True:
            record = pickle.load(file)
            if record[2]==Author:
                count+=1
    except EOFError:
        pass
    file.close()
    return count
```

```
def testProgram():
    while True:
        createFile()
        choice = input("Add more record (y/n)? ")
        if choice in 'Nn':
            break
    Author = input('Enter author name to search: ')
    n = countRec(Author)
    print("No of books are",n)

testProgram()
```

Output

```
Enter book number: 1002
Enter book Name: Graph Theory
Enter author: R. Singh
Enter price: 876.42
Add more record (y/n)? n
Enter author name to search: R. Singh
No of books are 1
```

10.13 A binary file "STUDENT.DAT" has structure (admission_number, Name, Percentage). Write a function count_rec() in Python that would read contents of the file"STUDENT.DAT" and display the details of those students whose percentage is above 75. Also display number of students scoring above 7514. Given a binary file employee.dat, created using dictionary object having keys: (empcode, name, and salary).

Code

```
import sys
import pickle
def createFile():
    file = open("student.dat","ab")
    admission_number = int(input("\nEnter date of admission(enter \'0\' for exit): "))
    Name = input("Enter student Name: ")
    Percentage = float(input("Enter percentage: "))
    record = [admission_number, Name, Percentage]
    pickle.dump(record, file)
    file.close()
```

```
def count_rec():
    file = open("student.dat","rb")
    count = 0
    try:
        while True:
            record = pickle.load(file)
            if record[2] > 75:
                print(record)
                count+=1
    except EOFError:
        pass
    print('No of students having more than 75% are', count)
    file.close()

while(1):
    print("1. Add data\n2. Percentages > 75%\n3. Exit")
    x=input("Enter choice\n")
    if x=="1":
        createFile()
    elif x=="2":
        count_rec()
    elif x=="3":
        sys.exit()
    else:
        print("Wrong choice")
```

Output

```
1. Add data
2. Percentages > 75%
3. Exit
Enter choice
1

Enter date of admission(enter '0' for exit): 1005
Enter student Name: H. Rao
Enter percentage: 76
1. Add data
2. Percentages > 75%
3. Exit
Enter choice
2
[101, 'Madhusudan', 89.0]
[104, 'Kiran', 90.0]
[1005, 'N. Sundaram', 93.0]
[1005, 'S. Mishra', 82.0]
```

```
[1005, 'H. Rao', 76.0]
No of students having more than 75% are 5
1. Add data
2. Percentages > 75%
3. Exit
Enter choice
3
```

10.14 Given a binary file employee.dat, created using dictionary object having keys: (empcode, name, and salary) Write a python function that add one more record at the end of file. Write a python function that display all employee records whose salary is more that 30000.

Code

```
import pickle

def add_record():
    file = open("employee.dat","ab")
    emp = {}
    emp['empcode'] = int(input("Enter employee code: "))
    emp['name'] = input("Enter employee Name: ")
    emp['salary'] = int(input("Enter salary: "))
    pickle.dump(emp, file)
    file.close()

def search():
    file = open("employee.dat","rb")
    try:
        while True:
            emp = pickle.load(file)
            if emp['salary']>30000:
                print(emp)
    except EOFError:
        pass
    file.close()

def testProgram():
    while True:
        add_record()
        choice = input("Add more record (y/n)? ")
        if choice in 'Nn':
            break
    print('Employee details having salary more than 30000')
    search()
```

```
testProgram()
```

Output

```
Enter employee code: 10002
Enter employee Name: Yusuf S
Enter salary: 80000
Add more record (y/n)? n
Employee details having salary more than 30000
{'empcode': 765001, 'name': 'R. Rao', 'salary': 45000}
{'empcode': 420001, 'name': 'B. Swami', 'salary': 90000}
{'empcode': 10002, 'name': 'Yusuf S', 'salary': 80000}
```

10.15 Write a function to search and display details of student whose rollno is '1005' from the binary file student.dat having structure [rollno, name, class and fees].

Code

```
import pickle
def search():
    file = open("student.dat","rb")
    try:
        while True:
            record = pickle.load(file)
            if record[0] == 1005:
                print(record)
    except EOFError:
        pass
    file.close()
search()
```

Output

```
[1005, 'N. Sundaram', 93.0]
[1005, 'G. Das', 71.0]
[1005, 'S. Mishra', 82.0]
[1005, 'H. Rao', 76.0]
```

10.16 . A binary file school.dat has structure(rollno, name, class, fees) Write a definition for function total_fees( ) that reads each object of file and calculate the total fees of students and display the same.

Code

```
import pickle
import sys
def insert_record():
    file = open("school.dat","ab")
    try:
        rollno=int(input("Enter rollno\n"))
        name = input("Enter name of student\n")
        class_name=input("Enter class\n")
        fees=float(input("Enter fees\n"))
        add=[rollno,name,class_name,fees]
        pickle.dump(add,file)
    except EOFError:
        pass
    file.close()

def total_fees():
    file = open("school.dat","rb")
    try:
        total = 0
        while True:
            record = pickle.load(file)
            total += record[3]
    except EOFError:
        pass
    print('Total Fees: ',total)
    file.close()

def show():
    file = open("school.dat","rb")
    try:
        while True:
            record = pickle.load(file)
            print(record)
    except:
        pass

while True:
    print("1. Add record\n2. Total Fess\n3. Show \n4.Exit")
    x=int(input("Enter your choice\n"))

    if x==1:
        insert_record()
    elif x==2:
        total_fees()
    elif x==3:
        show()
```

```
    elif x==4:
        sys.exit()
    else:
        print("Invalid option")
```

Output

```
1. Add record
2. Total Fess
3. Show
4.Exit
Enter your choice
1
Enter rollno
5052
Enter name of student
Gargie S
Enter class
B.Sc(Hons)
Enter fees
21000
1. Add record
2. Total Fess
3. Show
4.Exit
Enter your choice
3
[1001, 'Madhusudan', 'MCA', 13000.0]
[1002, 'Rahul', 'B.Tech', 28900.5]
[2004, 'Bishal Das', 'M.Tech', 45000.0]
[8001, 'Vijay M', 'B.Ed', 32000.0]
[5052, 'Gargie S', 'B.Sc(Hons)', 21000.0]
1. Add record
2. Total Fess
3. Show
4.Exit
Enter your choice
2
Total Fees:  139900.5
1. Add record
2. Total Fess
3. Show
4.Exit
Enter your choice
4
```

10.17 A binary file players.dat, containing records of following list format: [code, name, country and total runs] Write a python function that display all records where player name starts from 'A' Write a python function that accept country as an argument and count and display the number of players of that country. Write a python function that add one record at the end of file.

Code

```
'''A binary file players.dat, containing records of following list format: [code, name,
and total runs]
Write a python function that display all records where player name starts from 'A'
Write a python function that accept country as an argument and count and display the
number of players of that country.
Write a python function that add one record at the end of file.'''

import pickle

def createFile():
    file = open("players.dat","ab")
    Code = int(input("Enter player code: "))
    Name = input("Enter player Name: ")
    Country =input("Enter player country: ")
    Total_Runs = int(input("Enter total runs of player: "))
    record = [Code, Name, Country, Total_Runs]
    pickle.dump(record, file)
    file.close()

def search():
    file = open("players.dat","rb")
    count=0
    try:
        while True:
            record = pickle.load(file)
            if record[1][0] == 'A':
                count+=1
                print(record)
    except EOFError:
        pass
    file.close()
    return count

def countRec(Country):
    file = open("players.dat","rb")
    count = 0
```

```
    try:
        while True:
            record = pickle.load(file)
            print(record)
            if record[2]==Country:
                count+=1
    except EOFError:
        pass
    file.close()
    return count


def testProgram():
    while True:
        createFile()
        choice = input("Add more record (y/n)? ")
        if choice in 'Nn':
            break
    print("the number of players whose name starts with A:",search())
    Country = input('Enter country name to search: ')
    n = countRec(Country)
    print("No of players are",n)

testProgram()
```

Output

```
Enter player code: 3004
Enter player Name: Adam
Enter player country: Brazil
Enter total runs of player: 3200
Add more record (y/n)? n
[6001, 'Arun Kalyan', 'India', 4000]
[3004, 'Adam', 'Brazil', 3200]
the number of players whose name starts with A: 2
Enter country name to search: Brazil
[300001, 'Neymar', 'Brazil', 560]
[9001, 'El D Maria', 'Brazil', 320]
[2001, 'D Seliva', 'Brazil', 700]
[6001, 'Arun Kalyan', 'India', 4000]
[3004, 'Adam', 'Brazil', 3200]
No of players are 4
```

10.18 Given a binary file game.dat, containing records of following list format: [game_name, participants] Write a function in Python that would read contents from the file game.dat and creates a file named basket.dat copying only those records from game.dat where the game name is "Basket Ball"

Code

```
import pickle
import sys
def create_game_details():
    file = open("game.dat","ab")
    game_name=input("Enter Game Name\n")
    participants=input("Enter participant name\n")
    add=[game_name,participants]
    pickle.dump(add,file)
    file.close()
def show():
    print("Content of game.dat")
    infile=open("game.dat","rb")
    try:
        while True:
            record = pickle.load(infile)
            print(record)
    except EOFError:
        pass

    print("Content of basket.dat")
    infile=open("basket.dat","rb")
    try:
        while True:
            record = pickle.load(infile)
            print(record)
    except EOFError:
        pass
def countRec(country):

    infile = open("game.dat","rb")
    outfile = open("basket.dat","wb")
    outfile.seek(0)
    outfile.truncate()
    try:
        while True:
            record = pickle.load(infile)
            if record[0] == "Basket Ball":
```

```
                pickle.dump(record, outfile)
                print("Added to basket.dat!!!")
    except EOFError:
        pass
    infile.close()
    outfile.close()

while True:
    print("1. Add record\n2. Search Player Number\n3. Show\n4. Exit")
    x=int(input("Enter choice\n"))

    if x==1:
        create_game_details()
    elif x==2:
        c=input("Enter game Name\n")
        countRec(c)
    elif x==3:
        show()
    elif x==4:
        sys.exit()
    else:
        print("Wrong Choice")
```

Output

```
1. Add record
2. Search Player Number
3. Show
4. Exit
Enter choice
1
Enter Game Name
Cricket
Enter participant name
Viru
1. Add record
2. Search Player Number
3. Show
4. Exit
Enter choice
2
Enter game Name
Basket Ball
Added to basket.dat!!!
Added to basket.dat!!!
Added to basket.dat!!!
Added to basket.dat!!!
```

```
1. Add record
2. Search Player Number
3. Show
4. Exit
Enter choice
3
Content of game.dat
['Football', 'L Deo']
['Basket Ball', 'Reo N']
['Basket Ball', 'El Maria']
['Basket Ball', 'Vegan D']
['Basket Ball', 'Brayan']
['Cricket', 'Wasim']
['Cricket', 'Viru']
Content of basket.dat
['Basket Ball', 'Reo N']
['Basket Ball', 'El Maria']
['Basket Ball', 'Vegan D']
['Basket Ball', 'Brayan']
1. Add record
2. Search Player Number
3. Show
4. Exit
Enter choice
4
```