FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)^{TM}

HORMIS NAGAR, MOOKKANNOOR, ANGAMALY-683577

FOCUS ON EXCELLENCE

20MCA131 PROGRAMMING LAB LABORATORY RECORD

Name: AMALRAJ JOSEPH (FIT 21MCA-2016)

Branch: MASTER OF COMPUTER APPLICATIONS

Semester: 1   Batch: A   Roll No: 16

MARCH 2022

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)™

HORMIS NAGAR, MOOKKANNOOR, ANGAMALY-683577

FOCUS ON EXCELLENCE

CERTIFICATE

This is to certify that this is a Bonafide record of the Practical work done by ***AMALRAJ JOSEPH (FIT21MCA-2016)*** *in the* ***20MCA131 PROGRAMMING LAB*** *Laboratory towards the partial fulfilment for the award of the Master Of Computer Applications during the academic year 2021-2022.*

Signature of Staff in Charge   Signature of H O D

Name: Name:

Date of University practical examination ………………………

Signature of   Signature of
Internal Examiner   External Examiner

CONTENT

Sl No	Date of Experiment	Title of the Experiment	Page No:	Signature of Staff –In – Charge
1	28/10/2021	Display future leap years from current year to a final year entered by user.		
2	28/10/2021	List comprehensions: (a) Generate positive list of numbers from a given list of integers (b) Square of N numbers (c) Form a list of vowels selected from a given word (d) List ordinal value of each element of a word (Hint: use ord() to get ordinal values)		
3	28/10/2021	Count the occurrences of each word in a line of text.		
4	28/10/2021	Prompt the user for a list of integers. For all values greater than 100, store 'over' instead.		
5	10/11/2021	Store a list of first names. Count the occurrences of 'a' within the list		
6	10/11/2021	Enter 2 lists of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both		
7	10/11/2021	Get a string from an input string where all occurrences of first character replaced with '$', except first character. [eg: onion ->oni$n]		
8	10/11/2021	Create a string from given string where first and last characters exchanged. [eg: python ->nythop]		
9	10/11/2021	Accept the radius from user and find area of circle.		

Sl No	Date of Experiment	Title of the Experiment	Page No:	Signature of Staff –In – Charge
10	11/11/2021	Find biggest of 3 numbers entered.		
11	11/11/2021	Accept a file name from user and print extension of that.		
12	11/11/2021	Create a list of colors from comma-separated color names entered by user. Display first and last colors.		
13	11/11/2021	Accept an integer n and compute n+nn+nnn.		
14	11/11/2021	Print out all colors from color-list1 not contained in color-list2.		
15	17/11/2021	Create a single string separated with space from two strings by swapping the character at position 1.		
16	17/11/2021	Sort dictionary in ascending and descending order.		
17	17/11/2021	Merge two dictionaries.		
18	17/11/2021	Find gcd of 2 numbers.		
19	17/11/2021	From a list of integers, create a list removing even numbers.		
20	25/11/2021	Program to find the factorial of a number.		
21	25/11/2021	Generate Fibonacci series of N terms.		
22	25/11/2021	Find the sum of all items in a list		
23	25/11/2021	Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.		

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)[TM]

Sl No	Date of Experiment	Title of the Experiment	Page No:	Signature of Staff –In – Charge
24	02/12/2021	Display the given pyramid with step number accepted from user. Eg: N=4 1 2 4 3 6 9 8 12 16		
25	02/12/2021	Count the number of characters (character frequency) in a string.		
26	02/12/2021	Add 'ing' at the end of a given string. If it already ends with 'ing', then add 'ly'		
27	09/12/2021	Accept a list of words and return length of longest word.		
28	09/12/2021	Construct following pattern using nested loop *		
29	09/12/2021	Generate all factors of a number.		
30	29/01/2022	Create a package graphics with modules rectangle, circle and sub-package 3D-graphics with modules cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that finds area and perimeter of figures by different importing statements. (Include selective import of modules and import * statements)		

Sl No	Date of Experiment	Title of the Experiment	Page No:	Signature of Staff –In – Charge
31	13/01/2022	Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.		
32	13/01/2022	Create a Bank account with members account number, name, type of account and balance. Write constructor and methods to deposit at the bank and withdraw an amount from the bank.		
33	13/01/2022	Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.		
34	20/01/2022	Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time.		
35	20/01/2022	Create a class Publisher (name). Derive class Book from Publisher with attributes title and author. Derive class Python from Book with attributes price and no_of_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overriding.		
36	03/02/2022	Write a Python program to read a file line by line and store it into a list.		
37	03/02/2022	Write a Python program to read each row from a given csv file and print a list of string.		

COURSE OUTCOME 1

PROGRAM 1.1

Aim : Display future leap years from current year to a final year entered by user.

SOURCE CODE

```
current=2021
final=int(input("Enter the final limit :\t"))
for i in range(current,final):
        if(i%4==0):
                if(i%100==0):
                        if(i%400==0):
                                print(i)
                else:
                        print(i)
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p1$ python3 co1p1.py
Enter the final limit : 2050
2024
2028
2032
2036
2040
2044
2048
stud@debian:~/amalraj/python/Lab list/co1p1$ ▮
```

*****

PROGRAM 1.2

Aim : List comprehensions:

  (a) Generate positive list of numbers from a given list of integers.

  (b) Sqare of N numbers.

  (c) Form a list of vowels selected from a given word.

  (d) List ordinal value of each element of a word.

SOURCE CODE

```
sample_list=[]
size=int(input("Enter the size of list :\t"))
for i in range(0,size):
        sample_list.append(int(input("Enter the list element :\t")))
print("The positive elements are :")
positive_list=[]
j=0
for i in range(0,size):
        if(sample_list[i]>0):
                positive_list.append(sample_list[i])
                print(positive_list[j])
                j=j+1
print("The sqares are :")
for i in range (0,size):
        print(sample_list[i]**2)
vowel_list=['a','A','e','E','i','I','o','O','u','U']
vowels_in_string=[]
string=input("Enter a string :")
```

```
for i in string:

        for j in vowel_list:

                if(i==j):

                        vowels_in_string.append(i)

print("Vowels :-",vowels_in_string)

print("Ordinal values of ",string," are :")

for i in string:

        print(ord(i))
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p2$ gedit co1p2.py
stud@debian:~/amalraj/python/Lab list/co1p2$ python3 co1p2.py
Enter the size of list :        3
Enter the list element :        23
Enter the list element :        78
Enter the list element :        -56
The positive elements are :
23
78
The sqares are :
529
6084
3136
Enter a string :Amalraj
Vowels :- ['A', 'a', 'a']
Ordinal values of  Amalraj  are :
65
109
97
108
114
97
106
stud@debian:~/amalraj/python/Lab list/co1p2$ █
```

*****

PROGRAM 1.3

Aim : Count the occurence of each word in a line of text.

SOURCE CODE

```
list1=[]
ulist=[]
string=input("Enter a line of text :")
for i in string.split(" "):
        list1.append(i)
        if i not in ulist:
                ulist.append(i)
for i in ulist:
        print(i,"\t",list1.count(i))
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p4$ python3 co1p4.py
Enter a line of text :amal amal amal raj raj joseph
amal      3
raj       2
joseph    1
stud@debian:~/amalraj/python/Lab list/co1p4$
```

*****

PROGRAM 1.4

Aim : Prompt the user for a list of integers. For all values greater than 100, store 'over' instead.

SOURCE CODE

```
sample_list=[]
size=int(input("Enter the size :"))
for i in range(0,size):
        n=int(input("Enter the element :"))
        if(n<=100):
                sample_list.append(n)
        else:
                sample_list.append("over")
print(sample_list)
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p5$ gedit co1p5.py
stud@debian:~/amalraj/python/Lab list/co1p5$ python3 co1p5.py
Enter the size :5
Enter the element :100
Enter the element :23
Enter the element :190
Enter the element :200
Enter the element :34
[100, 23, 'over', 'over', 34]
stud@debian:~/amalraj/python/Lab list/co1p5$ ▌
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^(TM)

PROGRAM 1.5

Aim : Store a list of first names. Count the occurrence of 'a' within the list.

SOURCE CODE

```
print("\t\t\t\t\tCO1/06: PROGRAM TO COUNT A")
name_list=[]
count=0
n=int(input("Enter the number of names :"))
for i in range(0,n):
        name_list.append(input("Enter a name :"))
        for j in name_list[i]:
                if(j=='a'):
                        count=count+1
print("The count of 'a' is",count)
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p6$ python3 co1p6.py
                                CO1/06: PROGRAM TO COUNT A
Enter the number of names :5
Enter a name :Amal
Enter a name :Akash
Enter a name :Arjun
Enter a name :Kiran
Enter a name :Kumar
The count of 'a' is 4
stud@debian:~/amalraj/python/Lab list/co1p6$ 
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.6

Aim : Enter two list of integers. Check

    (a) Whether they are of same length.

    (b) Whether list sums to same value.

    (c) Whether any value occure in both.

SOURCE CODE

```
list1=[1,2,3,4,5]
list2=[6,7,8,9,10]
print("list1\t:",list1,"\nlist2\t:",list2)
if(len(list1)==len(list2)):
        print("Both list1 and list2 has same length.")
else:
        print("list1 and list2 have different sizes.")
if sum(list1)==sum(list2):
        print("Sum are same.")
else:
        print("Sum are different.")
flag=0
print("The common elements are :")
for i in list1:
        if i in list2:
                print(i)
                flag=1
if flag==0:
        print("No common elements!!!!")
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p7$ python3 co1p7.py
list1    : [1, 2, 3, 4, 5]
list2    : [6, 7, 8, 9, 10]
Both list1 and list2 has same length.
Sum are different.
The common elements are :
No common elements!!!!
stud@debian:~/amalraj/python/Lab list/co1p7$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.7

Aim : Get a string from an input string where all occurence of first charecter replaced with '$', except first character.

[onion -> oni$n]

SOURCE CODE

str=input("Enter a string :")

first=str[0]

str=str.replace(first,"$")

print("The new string is",first+str[1:])

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p8$ python3 co1p8.py
Enter a string :onion
The new string is oni$n
stud@debian:~/amalraj/python/Lab list/co1p8$ python3 co1p8.py
Enter a string :Onion
The new string is Onion
stud@debian:~/amalraj/python/Lab list/co1p8$ python3 co1p8.py
Enter a string :Amal
The new string is Amal
stud@debian:~/amalraj/python/Lab list/co1p8$ python3 co1p8.py
Enter a string :amal
The new string is am$l
stud@debian:~/amalraj/python/Lab list/co1p8$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.8

Aim : Create a string from given string where first and last characters exchanged.

[eg : Python ->nythoP]

SOURCE CODE

```
string=input("Enter a string :")
first=string[0]
last=string[-1]
length=len(string)
print("The new string is :",last+string[1:length-1]+first)
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p9$ gedit co1p9.py
stud@debian:~/amalraj/python/Lab list/co1p9$ python3 co1p9.py
Enter a string :Python
The new string is : nythoP
stud@debian:~/amalraj/python/Lab list/co1p9$ █
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.9

Aim : Accept the radius from user and find area of circle.

SOURCE CODE

```
pi=3.14
radius=int(input("Enter a number :"))
area=pi*(radius**2)
print("Area of the circle=",area)
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p10$ gedit co1p10.py
stud@debian:~/amalraj/python/Lab list/co1p10$ python3 co1p10.py
Enter a number :7
Area of the circle= 153.86
stud@debian:~/amalraj/python/Lab list/co1p10$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.10

Aim : Find biggest of 3 numbers entered .

SOURCE CODE

```
print("Enter 3 Numbers :")
a=int(input(""))
b=int(input(""))
c=int(input(""))
if (a>b)&(a>c):
        print(a,"is biggest")
if (b>a)&(b>c):
        print(b,"is biggest")
if (c>a)&(c>b):
        print(c,"is biggest")
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p11$ python3 co1p11.py
Enter 3 Numbers :
45
90
-76
90 is biggest
stud@debian:~/amalraj/python/Lab list/co1p11$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.11

Aim : Accept a file name from user and print extensin for that.

SOURCE CODE

```
import os
a=input("Enter the file name :")
print("Extension of file ",a,"is",os.path.splitext(a))
```

OUTPUT

```
stud@debian:~/amalraj/python/Lab list/co1p12$ gedit co1p12.py
stud@debian:~/amalraj/python/Lab list/co1p12$ python3 co1p12.py
Enter the file name :c01p12.py
Extension of file  c01p12.py is ('c01p12', '.py')
stud@debian:~/amalraj/python/Lab list/co1p12$ ▮
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.12

Aim : Create a list of coloures from comma-seperated colour names entered by user. Display first and last colours.

SOURCE CODE

```
clist=[]
cstring=input("Enter some elements separated by comma character :\n")
for i in cstring.split(","):
        clist.append(i)
print("First element :",clist[0],"\nLast element :",clist[-1])
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p13.py
Enter some elements separated by comma character :
black,blue,green,white,yellow
First element : black
Last element : yellow
stud@debian:~/Amalraj Joseph/Python$ 
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.13

Aim : Accept an integer n and compute n+nn+nnn.

SOURCE CODE

```
n=input("Enter a number :")
nn=n+n
nnn=nn+n
print("The sum is :",int(n)+int(nn)+int(nnn))
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p14.py
Enter a number :5
The sum is : 615
stud@debian:~/Amalraj Joseph/Python$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.14

Aim : Print out all colours from color list1 not contained in color list2.

SOURCE CODE

```
print("\t\t\tCO1 PROGRAM 15\n\n")
color_list1=["red","blue","green","yellow","white"]
color_list2=["red","blue","white","black"]
print("list1\t",color_list1,"\nlist2\t",color_list2,"\nThe colors in list1 not in list2 are :")
for i in color_list1:
        if i not in color_list2:
                print(i)
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p15.py
                        CO1 PROGRAM 15


list1    ['red', 'blue', 'green', 'yellow', 'white']
list2    ['red', 'blue', 'white', 'black']
The colors in list1 not in list2 are :
green
yellow
stud@debian:~/Amalraj Joseph/Python$ █
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.15

Aim : Create a single string separated with space from two strings by swapping the character at position 1.

 SOURCE CODE

```
string1="Fisat"
string2="Ankamaly"
f1=string1[0]
f2=string2[0]
string=f2+string1[1:]+" "+f1+string2[1:]
print("The new string is :",string)
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p16.py
The new string is : Aisat Fnkamaly
stud@debian:~/Amalraj Joseph/Python$
```

*****

PROGRAM 1.16

Aim : Sort dictinary in ascending and descending order.

SOURCE CODE

```
dict1={"a":1,"c":3,"d":2,"b":4}

l=list(dict1.items())

print(l)

l.sort()

print("Ascending Order is \n",l)

l=list(dict1.items())

l.sort(reverse=True)

print("Descending order is \n",l)
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p17.py
[('a', 1), ('c', 3), ('d', 2), ('b', 4)]
Ascending Order is
 [('a', 1), ('b', 4), ('c', 3), ('d', 2)]
Descending order is
 [('d', 2), ('c', 3), ('b', 4), ('a', 1)]
stud@debian:~/Amalraj Joseph/Python$ █
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.17

Aim : Merge two dictionaries.

SOURCE CODE

dic1={"Name":"Amal","Age":"20"}

dic2={"Qual":"PG","Gender":"M"}

dic1.update(dic2)

print(dic1)

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p18.py
{'Name': 'Amal', 'Age': '20', 'Qual': 'PG', 'Gender': 'M'}
stud@debian:~/Amalraj Joseph/Python$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.18

Aim : Find gcd of two numbers.

SOURCE CODE

```
n1=int(input("Enter the first number :"))
n2=int(input("Enter the second number :"))
if(n1>n2):
        small=n2
else:
        small=n1
for i in range(1,small+1):
        if((n1%i)==0 and (n2%i)==0):
                gcd=i
print("The GCD of",n1,"and",n2,"is",gcd)
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p19.py
Enter the first number :12
Enter the second number :45
The GCD of 12 and 45 is 3
stud@debian:~/Amalraj Joseph/Python$ ▮
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 1.19

Aim : Form a list of integers, create a list removing even numbers.

SOURCE CODE

```
list1=[]

list2=[]

n=int(input("Enter the list size :"))

for i in range(0,n):

        list1.append(int(input("Enter an element :")))

print("The list is\t",list1)

for i in list1:

        if i%2!=0:

                list2.append(i)

print("The odd list is\t",list2)
```

OUTPUT

```
stud@debian:~/Amalraj Joseph/Python$ python3 co1p20.py
Enter the list size :5
Enter an element :123
Enter an element :456
Enter an element :90
Enter an element :-456
Enter an element :23
The list is      [123, 456, 90, -456, 23]
The odd list is  [123, 23]
stud@debian:~/Amalraj Joseph/Python$ █
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

COURSE OUTCOME 2

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.1

Aim : Program to find the factorial of a number.

SOURCE CODE

```
n=int(input("Enter a Number :"))
factorial=1
for i in range(1,n+1):
        factorial=factorial*i
print("Factorial of",n,"=",factorial)
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ python3 p1.py
Enter a Number :5
Factorial of 5 = 120
stud@debian:~/amalraj/python/co2$
```

*****

PROGRAM 2.2

Aim : Generate Fibonacci series of N terms.

SOURCE CODE

```
n=int(input("Enter a Number :"))
print("The first",n,"fibonacci seriers is :")
f1=0
f2=1
for i in range(0,n):
        print(f1)
        f3=f1
        f1=f1+f2
        f2=f3
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ python3 p2.py
Enter a Number :11
The first 11 fibonacci seriers is :
0
1
1
2
3
5
8
13
21
34
55
stud@debian:~/amalraj/python/co2$ █
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.3

Aim : Find the sum of all items in a list.

SOURCE CODE

```
list1=[1,2,3,4,5,6,7]
summ=0
for i in list1:
        summ=summ+i
print("sum=",summ)
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p3.py
stud@debian:~/amalraj/python/co2$ python3 p3.py
sum= 28
stud@debian:~/amalraj/python/co2$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.4

Aim : Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect sqare.

SOURCE CODE

```
limit1=1000

limit2=9999

list1=[]

for i in range(limit1,limit2):

        j=i

        digit=[]

        while(i!=0):

                digit.append(i%10)

                i=int(i/10)

        count=0

        for n in digit:

                if n%2==0:

                        count=count+1

        if count==4:

                for k in range(31,100):

                        if((k**2)==j):

                                list1.append(j)

print(list1)
```

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

OUTPUT

```
^[[Astud@debian:~/amalraj/python/co2$ python3 p4.py
[4624, 6084, 6400, 8464]
stud@debian:~/amalraj/python/co2$ ▉
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.5

Aim : Display the given pyramid with step numbers accepted from user.

Eg : 4

```
     1

     2      4

     3      6      9

     4      8      12     16
```

SOURCE CODE

```python
n=int(input("Enter a number :"))
for i in range(1,n+1):
        for j in range(i,(i*i)+1,i):
                print(j,"\t",end="")
        print("\n")
```

OUTPUT

```
Enter a number :5
1

2       4

3       6       9

4       8       12      16

5       10      15      20      25

stud@debian:~/amalraj/python/co2$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.6

Aim : Count the number of characters (character frequency) in a string.

SOURCE CODE

```
string=input("Enter a string :")
ulist=[]
for i in string:
        if i not in ulist:
                ulist.append(i)
for i in ulist:
        count=0
        for j in string:
                if(i==j):
                        count+=1
        print(i,"\t:",count)
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p6.py
stud@debian:~/amalraj/python/co2$ python3 p6.py
Enter a string :amalraj joseph
a        : 3
m        : 1
l        : 1
r        : 1
j        : 2
         : 1
o        : 1
s        : 1
e        : 1
p        : 1
h        : 1
stud@debian:~/amalraj/python/co2$
```

*****

PROGRAM 2.7

Aim : Add 'ing' at the end of a given string.If it already ends with 'ing' , then add 'ly'.

SOURCE CODE

```
string=input("Enter a string :")
if(string[-3:]=="ing"):
        string+="ly"
else:
        string+="ing"
print(string)
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p7.py
stud@debian:~/amalraj/python/co2$ python3 p7.py
Enter a string :India is my country
India is my countrying
stud@debian:~/amalraj/python/co2$ python3 p7.py
Enter a string :he is running
he is runningly
stud@debian:~/amalraj/python/co2$ 
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.8

Aim : Accept a list of words and return length of longest word.

SOURCE CODE

```
wlist=[]
print("Enter 5 words :")
for i in range(0,5):
        wlist.append(input(""))
temp=wlist[0]
for i in range(1,5):
        if len(wlist[i])>len(temp):
                temp=wlist[i]
print("Length of longest word is",len(temp))
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p8.py
stud@debian:~/amalraj/python/co2$ python3 p8.py
Enter 5 words :
amal
ama
am
a
amalraj
Length of longest word is 7
stud@debian:~/amalraj/python/co2$ █
```

*****

PROGRAM 2.9

Aim : Construct following pattern using nested loop.

```
*
**
***
****
*****
****
***
**
*
```

SOURCE CODE

```python
for i in range(1,6):
        for j in range(0,i):
                print("*",end="")
        print("\n")
for i in range(4,0,-1):
        for j in range(0,i):
                print("*",end="")
        print("\n")
```

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p9.py
stud@debian:~/amalraj/python/co2$ python3 p9.py
*

**

***

****

*****

****

***

**

*
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 2.10

Aim : Generate all factors of a number.

SOURCE CODE

```
n=int(input("Enter a number :"))
print("The factors are :")
for i in range(1,n+1):
        if(n%i)==0:
                print(i)
```

OUTPUT

```
stud@debian:~/amalraj/python/co2$ gedit p10.py
stud@debian:~/amalraj/python/co2$ python3 p10.py
Enter a number :12
The factors are :
1
2
3
4
6
12
stud@debian:~/amalraj/python/co2$ 
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

COURSE OUTCOME 3

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 3.1

Aim : Create a package graphics with modules rectangle,circle and sub-package 3D graphics with module cuboid and sphere. Include methods to find area and perimeter of respective figures in each modules. Write programs that finds area and perimeter of figures by different importing statements.

Terminal Commands

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$ mkdir graphics
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$ cd graphics
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics$ gedit __init__.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics$ gedit rectangle.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics$ gedit circle.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics$ mkdir tdgraphics
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics$ cd tdgraphics
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics/tdgraphics$ gedit __init__.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics/tdgraphics$ gedit cuboid.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics/tdgraphics$ gedit sphere.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3/graphics/tdgraphics$ 
```

graphics\rectangle.py

```python
class Rectangle:

        def __init__(self,length,width):

                self.length=length

                self.width=width

        def area(self):

                return (self.length*self.width)

        def perimeter(self):

                return (2*(self.length+self.width))
```

graphics\circle.py

```python
global pi

pi=3.1416

class Circle:
```

```
        global pi
        pi=3.1416
        def __init__(self,radius):
                self.radius=radius
        def area(self):
                return (pi*(self.radius**2))
        def perimeter(self):
                return (2*pi*self.radius)
```

graphics\tdgraphics\sphere.py

```
global pi
pi=3.1416
class Sphere:
        def __init__(self,radius):
                self.radius=radius
        def volume(self):
                r=self.radius
                return ((4/3)*pi*(r**3))
        def area(self):
                r=self.radius
                return (4*pi*(r**2))
```

graphics\tdgraphics\cuboid.py

```
class Cuboid:
        def __init__(self,length,width,height):
                self.l=length
                self.w=width
                self.h=height
        def volume(self):
```

```
                return (self.l*self.w*self.h)
        def area(self):
        #method to find total surface area
                l=self.l
                w=self.w
                h=self.h
                return (2*((l*w)+(w*h)+(l*h)))
```

Program1.py

```python
from graphics import rectangle as rt
from graphics import circle
from graphics.tdgraphics import *


#Rectangle
r=rt.Rectangle(10,12)
print("_____RECTANGLE_____")
print("length =",r.length)
print("width =",r.width)
print("area=",r.area())
print("perimeter=",r.perimeter())


#Circle
c=circle.Circle(12)
print("_____CIRCLE_____")
print("radius =",c.radius)
print("area=",c.area())
```

```
print("perimeter=",c.perimeter())


#Sphere

s=sphere.Sphere(12)

print("_____SPHERE_____")

print("radius =",s.radius)

print("area=",s.area())

print("volume=",s.volume())




#Cuboid

cu=cuboid.Cuboid(13,11,14)

print("_____CUBOID_____")

print("length =",cu.l)

print("width =",cu.w)

print("height =",cu.h)

print("area=",cu.area())

print("volume=",cu.volume())
```

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$ gedit program1.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$ python3 program1.py
_____RECTANGLE_____
length = 10
width = 12
area= 120
perimeter= 44
_____CIRCLE_____
radius = 12
area= 452.3904
perimeter= 75.3984
_____SPHERE_____
radius = 12
area= 1809.5616
volume= 7238.246399999999
_____CUBOID_____
length = 13
width = 11
height = 14
area= 958
volume= 2002
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$ ^C
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO3$
```

*****

COURSE OUTCOME 4

PROGRAM 4.1
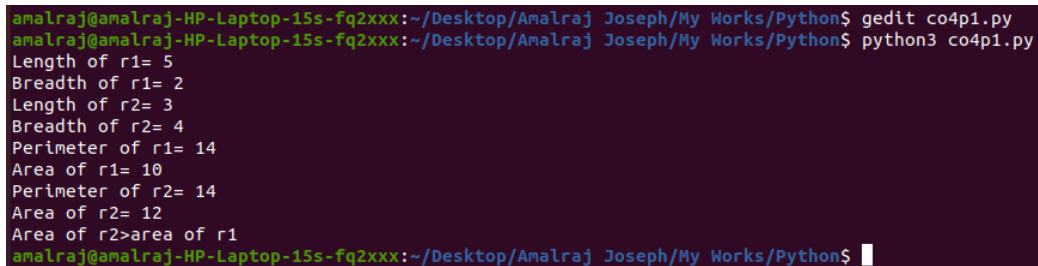
Aim : Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two rectangle objects by their area.

SOURCE CODE

```
class Rectangle:
        def __init__(self,l,b):
                self.length=l
                self.breadth=b
        def area(self):
                return self.length*self.breadth
        def perimeter(self):
                return 2*(self.length+self.breadth)
r1=Rectangle(5,2)
r2=Rectangle(3,4)
a1=r1.area()
a2=r2.area()
print("Length of r1=",r1.length)
print("Breadth of r1=",r1.breadth)
print("Length of r2=",r2.length)
print("Breadth of r2=",r2.breadth)
print("Perimeter of r1=",r1.perimeter())
print("Area of r1=",a1)
print("Perimeter of r2=",r2.perimeter())
print("Area of r2=",a2)
```

```python
if(a1>a2):

        print("Area of r1>area of r2")

elif(a2>a1):

        print("Area of r2>area of r1")

else:

        print("Area of r1=area of r2")
```

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ gedit co4p1.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ python3 co4p1.py
Length of r1= 5
Breadth of r1= 2
Length of r2= 3
Breadth of r2= 4
Perimeter of r1= 14
Area of r1= 10
Perimeter of r2= 14
Area of r2= 12
Area of r2>area of r1
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$
```

*****

PROGRAM 4.2
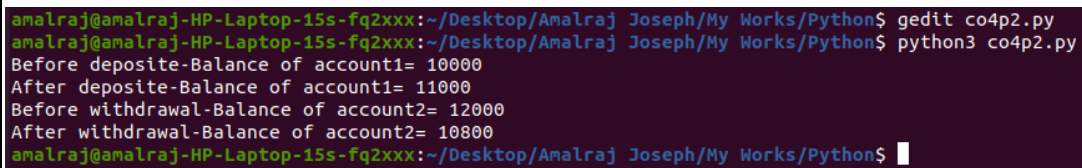
Aim : Create a Bank account with members account number, name, type of account and balance. Write constructor and methods to deposite at the bank and withdraw an amount from the bank.

SOURCE CODE

```
class Bank_account:
        def __init__(self,ano,name,type,balance):
                self.account_number=ano
                self.name=name
                self.type_of_account=type
                self.balance=balance
        def deposite(self,amount):
                self.balance=self.balance+amount
        def withdraw(self,amount):
                if(amount>self.balance):
                        print("Insufficient Balance!!!")
                else:
                        self.balance=self.balance-amount
account1=Bank_account(101,"Amalraj Joseph","Savings",10000)
account2=Bank_account(102,"Abhinav H","Savings",12000)
#Depositing Rs 1000 to account1
print("Before deposite-Balance of account1=",account1.balance)
account1.deposite(1000)
print("After deposite-Balance of account1=",account1.balance)
```

#Withdrawing Rs 1200 from account2

print("Before withdrawal-Balance of account2=",account2.balance)

account2.withdraw(1200)

print("After withdrawal-Balance of account2=",account2.balance)

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ gedit co4p2.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ python3 co4p2.py
Before deposite-Balance of account1= 10000
After deposite-Balance of account1= 11000
Before withdrawal-Balance of account2= 12000
After withdrawal-Balance of account2= 10800
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ 
```
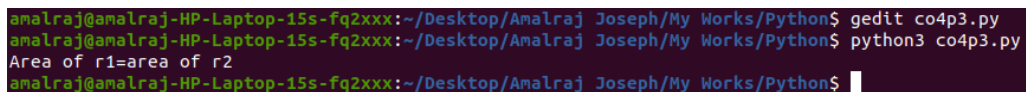
*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 4.3

Aim : Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of two rectangles.

SOURCE CODE

```
class Rectangle:
        def __init__(self,l,b):
                self.__length=l
                self.__width=b
        def __lt__(self,ob):
                if((self.__length*self.__width)<(ob.__length*ob.__width)):
                        return True
                else:
                        return False
r1=Rectangle(12,2)
r2=Rectangle(6,4)
if(r1<r2):
        print("Area of r1<area of r2")
elif(r2<r1):
        print("Area of r2<area of r1")
else:
        print("Area of r1=area of r2")
```

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ gedit co4p3.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ python3 co4p3.py
Area of r1=area of r2
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ 
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 4.4

Aim : Create a class Time with private attributes hour,minute and second. Overload '+' operator to find sum of two time.

SOURCE CODE

```
class Time:
        def __init__(self,h,m,s):
                self.__hour=h
                self.__minute=m
                self.__second=s
        def __add__(self,ob):
                hour=self.__hour+ob.__hour
                minute=self.__minute+ob.__minute
                second=self.__second+ob.__second
                t=Time(hour,minute,second)
                return t
        def print_it(self):
                print("Hour :",self.__hour)
                print("Minute :",self.__minute)
                print("Second :",self.__second)
t1=Time(10,10,10)
t2=Time(20,20,20)
t3=t1+t2
t3.print_it()
```

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ gedit co4p4.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ python3 co4p4.py
Hour : 30
Minute : 30
Second : 30
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 4.5

Aim : Create a class Publisher(name). Derive class Book from Publisher with attributes title and author. Derive class python from Book with attributes price and no_of_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overreading.

SOURCE CODE

```
class Publisher:
        def __init__(self,name):
                self.name=name
class Book(Publisher):
        def __init__(self,name,title,auther):
                super().__init__(name)
                self.title=title
                self.auther=auther
        def print_function(self):
                print("This Fuction is a member fuction of class Publisher")
class Python(Book):
        def __init__(self,name,title,auther,price,nop):
                super().__init__(name,title,auther)
                self.price=price
                self.nop=nop
        def print_function(self):
                print("Name :",self.name)
                print("Title :",self.title)
                print("Auther :",self.auther)
                print("Price :",self.price)
                print("Number of Pages :",self.nop)
```
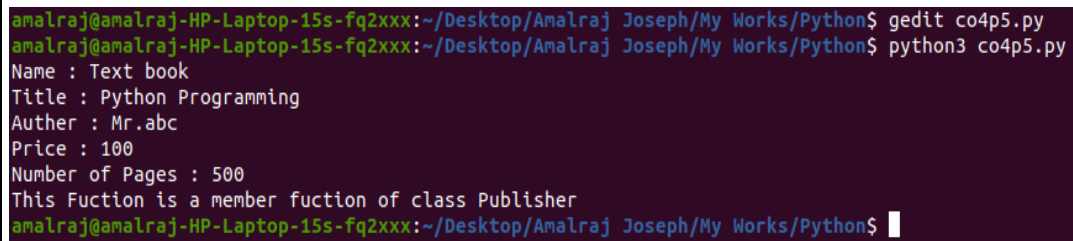
p1=Python("Text book","Python Programming","Mr.abc",100,500)

p1.print_function()

p2=Book("a","b","c")

p2.print_function()

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ gedit co4p5.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ python3 co4p5.py
Name : Text book
Title : Python Programming
Auther : Mr.abc
Price : 100
Number of Pages : 500
This Fuction is a member fuction of class Publisher
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python$ 
```

*****

COURSE OUTCOME 5

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 5.1

Aim : Write a program to read a file line by line and store it into a list

SOURCE CODE

```
file=open("text.txt","r")
lines=[]
for line in file:
        lines.append(line.strip())
print(lines)
```

text.txt

computer science, the study of computers and computing, including their theoretical and algorithmic foundations, hardware and software, and their uses for processing information. The discipline of computer science includes the study of algorithms and data structures, computer and network design, modeling data and information processes, and artificial intelligence. Computer science draws some of its foundations from mathematics and engineering and therefore incorporates techniques from areas such as queueing theory, probability and statistics, and electronic circuit design. Computer science also makes heavy use of hypothesis testing and experimentation during the conceptualization, design, measurement, and refinement of new algorithms, information structures, and computer architectures.

OUTPUT

```
analraj@analraj-HP-Laptop-15s-fq2xxx:~/Desktop/Analraj Joseph/My Works/Python/CO4$ gedit co4p1.py
analraj@analraj-HP-Laptop-15s-fq2xxx:~/Desktop/Analraj Joseph/My Works/Python/CO4$ python3 co4p1.py
['computer science, the study of computers and computing, including their theoretical and algorithmic foundations, hardware and software, and their uses for processing information. The discipline of comp
uter science includes the study of algorithms and data structures, computer and network design, modeling data and information processes, and artificial intelligence. Computer science draws some of its fo
undations from mathematics and engineering and therefore incorporates techniques from areas such as queueing theory, probability and statistics, and electronic circuit design. Computer science also makes
heavy use of hypothesis testing and experimentation during the conceptualization, design, measurement, and refinement of new algorithms, information structures, and computer architectures.']
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}

PROGRAM 5.2

Aim : Write a Python program to read each row from a given csv file and print a list of strings.

SOURCE CODE

```
import csv
with open("text.csv","r") as file:
        reader=csv.reader(file)
        for row in reader:
                print(row)
```

text.csv

```
Id,Name,Desig,Salary
001,Arun,Manager,100000
002,Anu,Secretary,40000
003,Akash,Security,30000
```

OUTPUT

```
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO4$ gedit text.csv
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO4$ gedit co4p2.py
amalraj@amalraj-HP-Laptop-15s-fq2xxx:~/Desktop/Amalraj Joseph/My Works/Python/CO4$ python3 co4p2.py
['Id', 'Name', 'Desig', 'Salary']
['001', 'Arun', 'Manager', '100000']
['002', 'Anu', 'Secretary', '40000']
['003', 'Akash', 'Security', '30000']
```

*****

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY(FISAT)^{TM}