FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)™

HORMIS NAGAR, MOOKKANNOOR

ANGAMALY-683577

'FOCUS ON EXCELLENCE'

LABORATORY RECORD

20MCA131 - PROGRAMMING LAB

Name : EMMANUEL JOSE

Branch : MASTER OF COMPUTER APPLICATION

Semester : 1 Batch: 2021 A Roll No: 56

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 ***Emmanuel Jose (FIT21MCA-2056)*** *in the* ***20MCA131 PROGRAMMING*** *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

Name:

Date:

Signature of H.O.D

Name:

Date of University practical examination ……

Signature of

Internal Examiner

Signature of

External Examine

CONTENT

12	**11-11-21**	Create a list of colors from comma-separated color names entered by user. Display first and last colors.	**8**	
13	**11-11-21**	Accept an integer n and compute n+nn+nnn	**9**	
14	**11-11-21**	Print out all colors from color-list1 not contained in color-list2.	**9**	
15	**11-11-21**	Create a single string separated with space from two strings by swapping the character at position 1	**10**	
16	**17-11-21**	Sort dictionary in ascending and descending order.	**10**	
17	**17-11-21**	Merge two dictionaries	**11**	
18	**17-11-21**	Find gcd of 2 numbers.	**11**	
19	**17-11-21**	From a list of integers, create a list removing even numbers.	**12**	
20	**17-11-21**	Program to find the factorial of a number	**13**	
21	**25-11-21**	Generate Fibonacci series of N terms.	**13**	
22	**25-11-21**	Find the sum of all items in a list	**14**	
23	**25-11-21**	Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square	**15**	
24	**25-11-21**	Display the given pyramid with step number accepted from user. Eg: N=4 1 2 4 3 6 9 8 12 16	**16**	
25	**02-12-21**	Count the number of characters (character frequency) in a string	**17**	

26	**02-12-21**	Add 'ing' at the end of a given string. If it already ends with 'ing', then add 'ly'	**18**	
27	**02-12-21**	Accept a list of words and return length of longest word.	**19**	
28	**09-12-21**	Construct following pattern using nested loop *	**20**	
29	**09-12-21**	Generate all factors of a number	**22**	
30	**09-12-21**	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)	**23**	
31	**13-01-22**	Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.	**25**	
32	**13-01-22**	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	**26**	
33	**13-01-22**	Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.	**27**	
34	**20-01-22**	Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time.	**28**	

35	**20-01-22**	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.	**29**	
36	**03-02-22**	Write a Python program to read a file line by line and store it into a list.	**30**	
37	**03-02-22**	Write a Python program to read each row from a given csv file and print a list of string.	**30**	

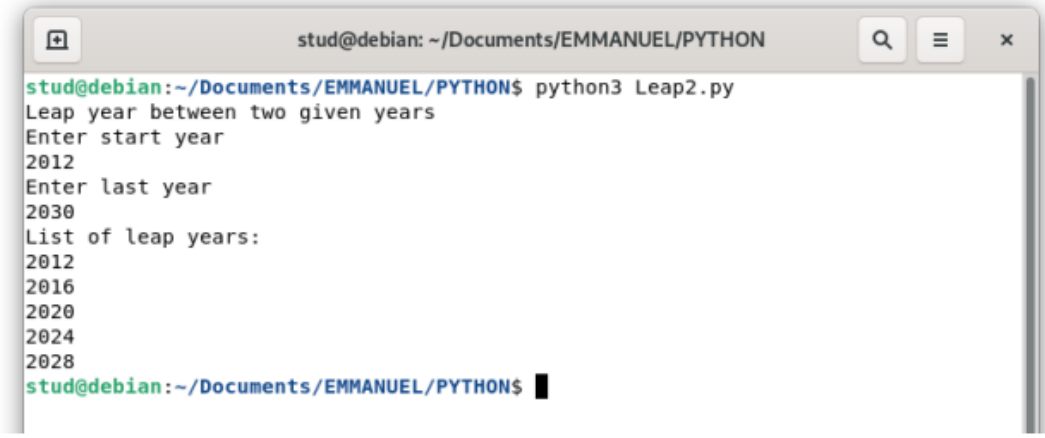
COURSE OUTCOME 1

CO1 Q-1

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

```
print("Print leap year between two given years")
print("Enter start year")
start Year=int(input())
print("Enter last year")
end Year=int(input())
print("List of leap years:")
for year in range(start Year, end Year):
if(0==year%4)and(0!=year%100)or(0==year%400):
        print(year)
```

Output:



CO2 Q-2

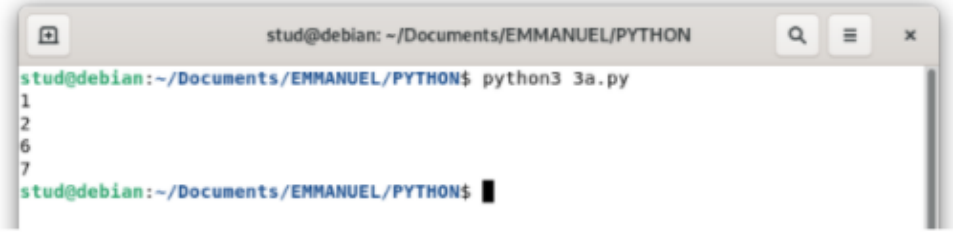
List Comprehensions:

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

Program Code:

```
list=[1,-2,67,45,-5]
for num in list:
        if num>0:
            print(num)
```

Output:

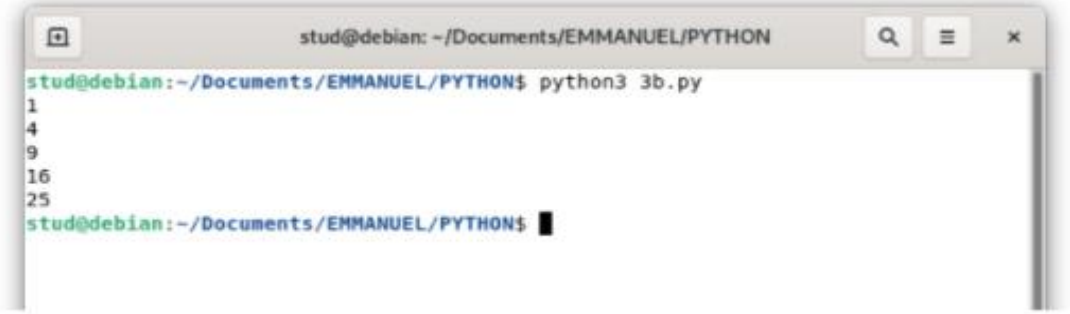


b) Square of N numbers.

Program code:

```
list=[5,8,-1,-2]
for num in list:
        print(num * num)
```

Output:

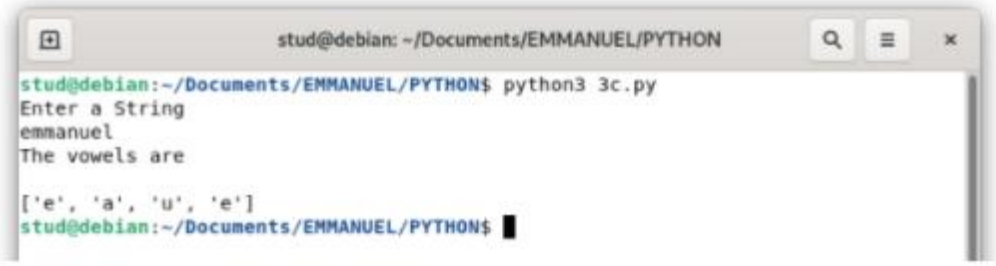


c)Form a list of vowels selected from a given word.

Program Code:

```
stringA="emmanuel"
print("Given String:\n",stringA)
vowels="AaEeIiOoUu"
li=[]
for r in stringA:
if r in vowels:
li.append(r)
        print(li)
```

Output:



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

Program code:

```
stringp="ABCD"
'''for c in stringp:
print(ord(c))'''
s=[ord(p)for p in stringp]
        print(s)
```

Output:

CO1 Q-3

Count the occurrences of each word in a line of text.

Program code:

```
s="HI hello,  i am mia and i am dia"
l=s.split()
d={x:l.count(x)  for x in l}
print(d)
```

Output:

```
{'HI': 1, 'hello,': 1, 'i': 2, 'am': 2, 'mia': 1, 'and': 1, 'dia': 1}
```

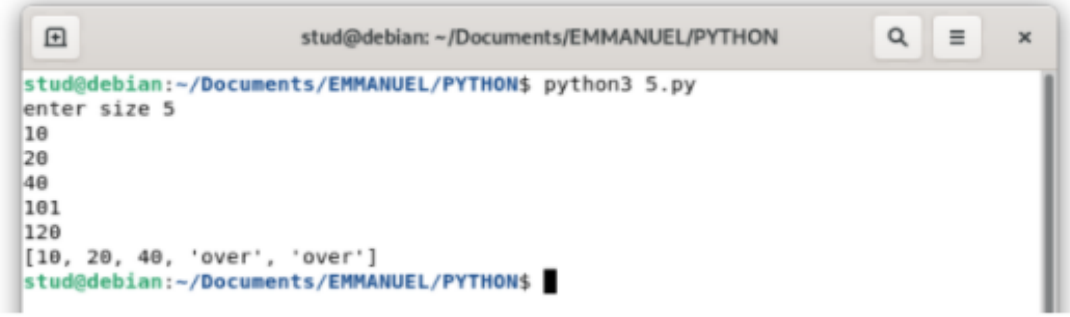
CO1 Q-4

Prompt the user for a list of integers. For all values greater than 100 store "over" instead.

Program code:

```
x=int(input("enter  size"))
li=[]
foriinrange(0,x):
el=int(input())
if(el>=100):
li.append("over")
else:
li.append(el)
        print(li)
```

Output:



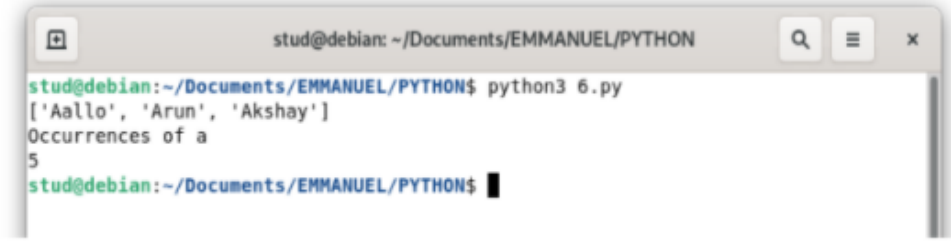
CO1 Q-5

Store the list of first names. Count the occurrence of "a" within the list

Program code:

```
list=['Aallo','Arun','Akshay']
count=0
print(list)
foriinlist:
forkini:
if(k=='a'):
count=count+1
        print(count)
```

Output:



CO1 Q-6

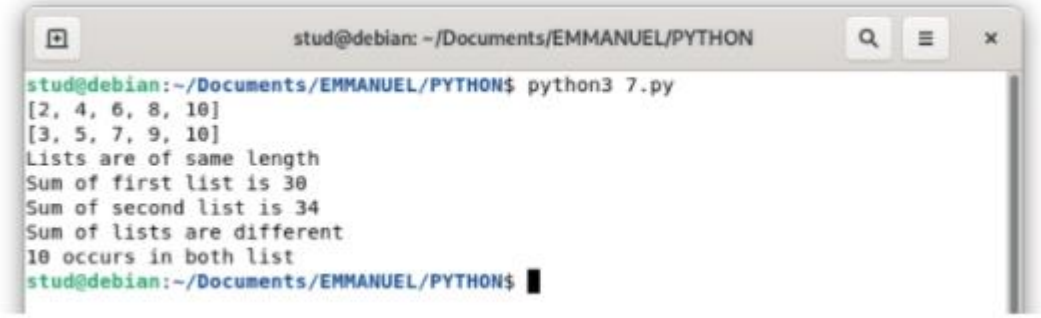
Enter 2 list of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both.

Program code:

```
l1=[2,4,6,8,10]
l2=[3,5,7,9,10]
print(l1)
print(l2)
if len(l1)==len(l2):
print("Lists are of same length")

else:
print("Lists are of different length")
s1=0
s2=0
for i in range(len(l1)):
s1=s1+l1[i]
print("Sum of first list is",s1)
for j in range(len(l2)):
s2=s2+l2[j]
print("Sum of second list is",s2)
if (s1==s2):
print("Sum of lists is same")
else:
print("Sum of lists are different")
for i in l1:
if i in l2:
print(i,"occurs in both list")
```

Output:



CO1 Q-7

Get a string from an input string where all occurrences of first character replaced with "$', except first character.
[eg:onion->oniSn]

Program code:

```
s=input("enter a string\n")
print("entered string is:",s)
a=s[0]
str=s.replace(a,"$")
strl=a+str[1:]
print(strl
```

Output:

CO1 Q-8

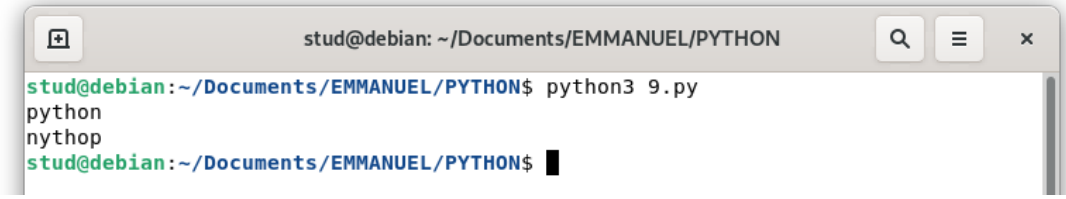
Create a string from given string where first and last characters exchanged.

[eg:python>nythop].

Program code:

```
a=input("enter a string")
print(a)
a1=a[0]
a2=a[-1]
print(a1)
print(a2)
rev=(a2+a[1:len(a)-1]+a1)
print(rev)
```

Output:



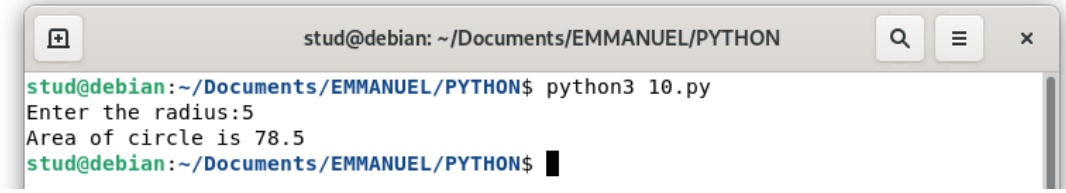
CO1 Q-9

Accept the radius from user and find area of circle.

Program code:

```
p=int(input("enter  the radius"))
a=3.14*p*p
print(a)
```

Output:



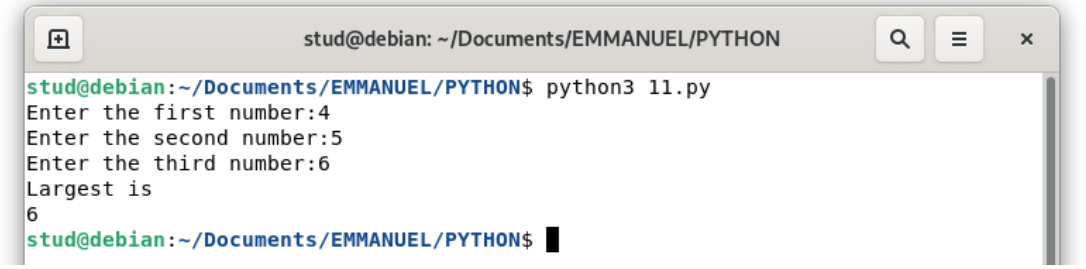
CO1 Q-10

Find biggest of 3 numbers entered.

Program code:

```
a=int(input("enter  ist number"))
b=int(input("enter  2nd number"))
c=int(input("enter  3rd number"))
ifa>b and a>c:
print(a)
ifc>b and c>a:
print(c)
else:
        print(b)
```

Output:

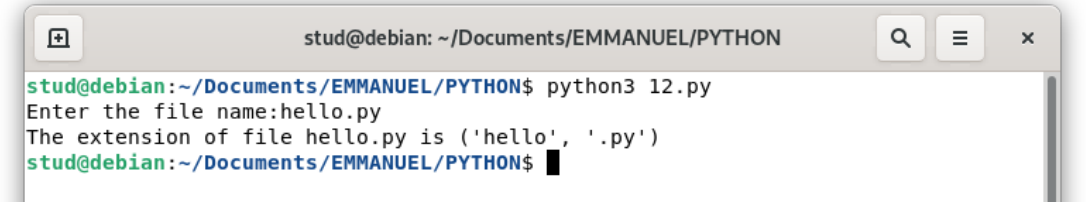


CO1 Q-11

Accept a filename from user and print extension of that.
Program code:

```
Import os
a=input("Enter the filename\n")
print(os.path.splitext(a))
```

Output:



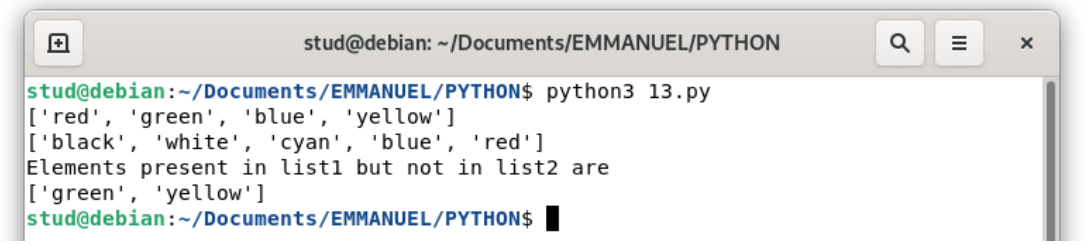
CO1 Q -12

Create a list of colors from comma-separated color names entered by user . Display First and last colors.

Program code:

```
list1=[]
string=input("Enter colors separated by comma:\n")
for I in string.split(","):
list1.append(i)
print("First and last colors in the list are",list1[0],"and",list1[-1])
```

Output:



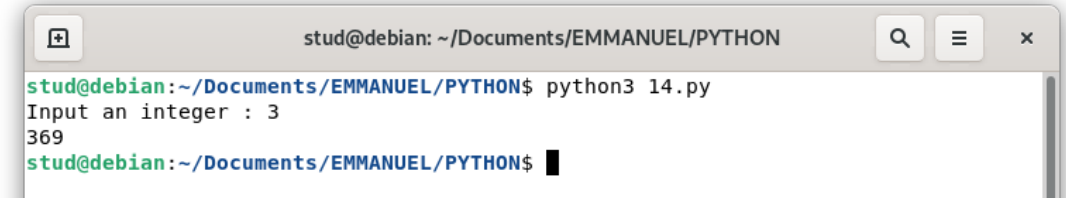
CO1 Q -13

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

Program code:

```
a=int(input("Input an integer:"))
n1=(a*1)
n2=(a*11)
n3=(a*111)
print(n1+n2+n3)
```

Output:



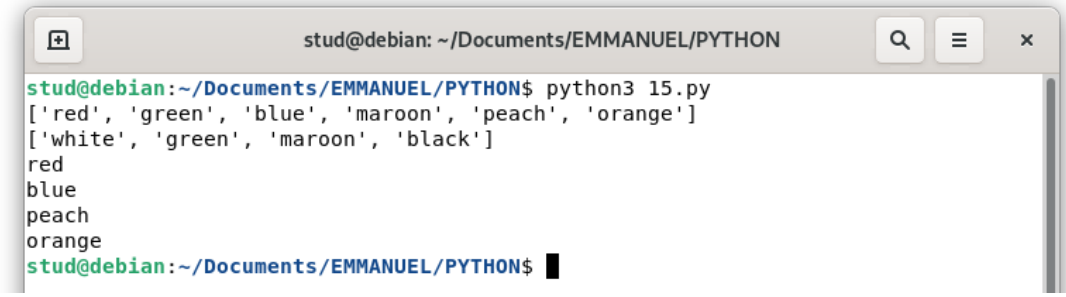
CO1 Q -14

Print out all colors from color- list1 not contained in color-list2.

Program code:

```
list1=["red","green","blue","yellow"]
list2=["black","white","cyan","blue","red"]
l3=[]
print(list1)
print(list2)
foriinlist1:
ifinotinlist2:
l3.append(i)
print("Elements present in list1 but not in list2 are")
print(l3)
```

Output:



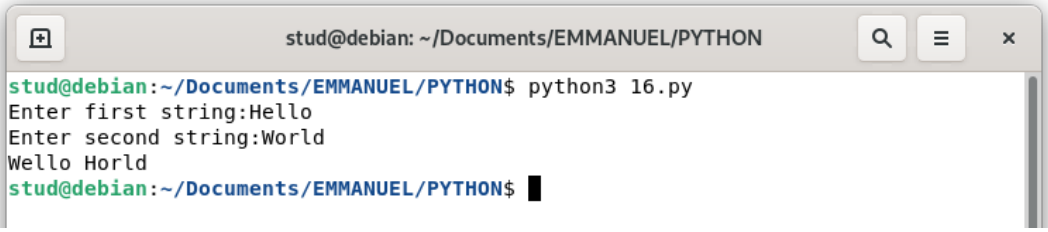
CO1 Q -15

Create a single string separated with space from 2 strings swapping the character at position.

Program code:

```
str1=input("Enter string1:")
str2=input("Enter string2:")
temp=str1[0]
str1=str1.replace(str1[0],str2[0])
str2=str2.replace(str2[0],temp)
str=str1+" "+str2
print(str)
```

Output:



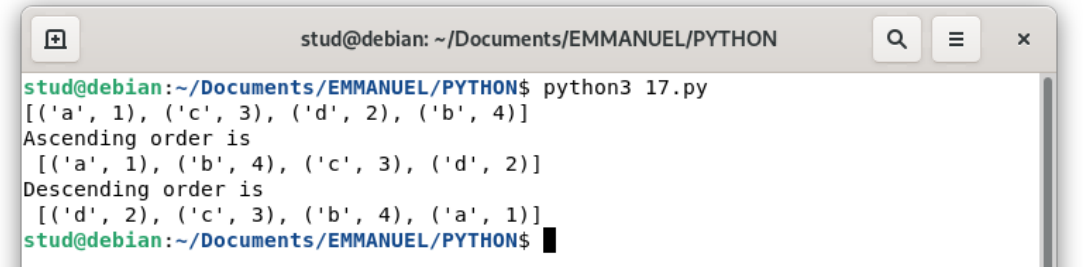
CO1 Q -16

Sort a dictionary in ascending and descending order

Program code:

```
d={1:2,3:4,4:3,2:1,0:0}
list1=list(d.items())
dict=dict(list1)
print("Dictionary=",dict)
list1.sort()
print('Ascending order is',list1)
list1=list(d.items())
list1.sort(reverse=True)
print('Descending order is',list1)
```

Output:



CO1 Q -17

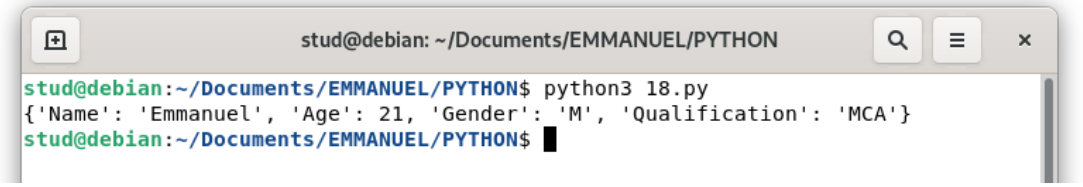
```
dict1={"Name":"Emmanuel","Age":21}

dict2={"Gender":"M","Qualification":"MCA"}

dict1.update(dict2)

print(dict1)
```

Output:



PROGRAM-18

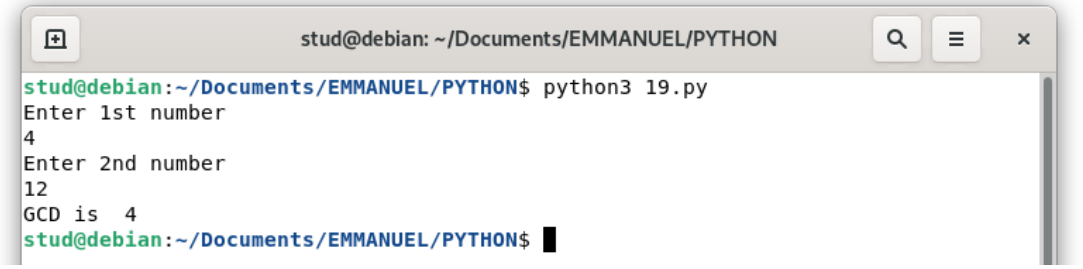
Find gcd of 2 numbers

Program code:

```
a=int(input("enter  1st number"))

b=int(input("enter  2nd number"))
z=min(a,b)
for I in range(1,z+1):
if((a % i== 0)and(b % i==0)):
gcd=i
print("gcd  is=",gcd)
```

Output:



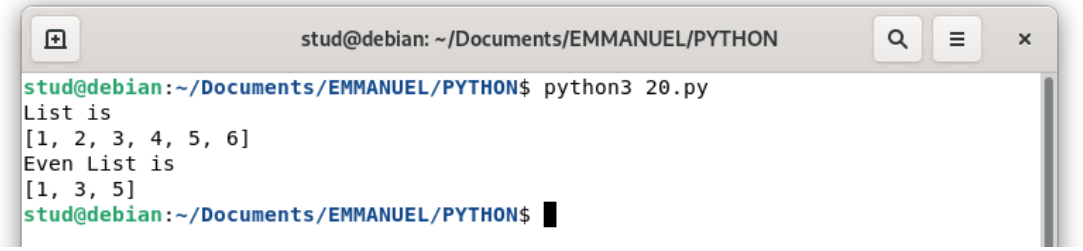
PROGRAM-19

From a list of integers, create a list removing even numbers.

Program code:

```
list=[12,13,14,15,16,21]
l1=[]
print(list)
print("New list")
for i in list:
    if i%2!=0:
        l1.append(i)
print(l1)
```

Output:



COURSE OUTCOME 2

CO2 Q -20

Program to find the factorial of a number

Program code:

```
fact=1
n=int(input('enter the value'))
for i in range(1,n+1):
        fact=fact*i
print(fact)
```

Output:

```
n=int(input("Enter a number:"))
fact=1
for i in range(1,n+1):
  fact=fact*i
print(fact)
```

```
Enter a number:5
120
```

CO2 Q -21

Generate Fibonacci series of N terms

Program code:

```
f1=0
f2=1
n=int(input('enter the number))
print(f1)
print(f2)
for i in range(2,n):
        f3=f1+f2
        print(f3)
        f1=f2
        f2=f3
```

Output:

```
n=int(input("Enter a number:"))
f1=0
f2=1
print(f1)
print(f2)
for i in range(0,n-2):
  f3=f1+f2
  print(f3)
  f1=f2
  f2=f3
```

```
Enter a number:5
0
1
1
2
3
```

CO2 Q --22

Find the sum of all items in list?

Program code:

```
list=[1,2,3,4,5,6,7,8,9,10]
sum=0
for i in list:
   sum=sum +int(i)
print("sum:" ,sum)
```

Output:

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

```
15
```

CO2 Q --23

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

Program 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(k)
print(list1)
```

Output:

```
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(k)
print(list1)
```

```
68
78
80
92
[4624, 6084, 6400, 8464]
```

CO2 Q --24

Display the given pyramid with step number accepted from user.

Program code:

```
l=int(input('Enter  the limit:'))
for i in range(1,l+1):
    for j in range(1,i+1):
        c=i*j
        print(c,end=" ")
    print("\n")
```

Output:

```
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")
```

```
Enter a number:5
1

2       4

3       6       9

4       8       12      16

5       10      15      20      25
```

CO2 Q --25

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

Program code:

```
string=input("Enter  a string:")
list1=[]
for i in string:
   if i not in list1:
    list1.append(i)
for i in list1:
   count=0
   for j in string:
 if(i==j):
   count=count+1
 print(i,"\t:",count)
```

Output:

```
string=input("Enter a string:")
list1=[]
for i in string:
  if i not in list1:
    list1.append(i)
for i in list1:
  count=0
  for j in string:
    if(i==j):
      count=count+1
  print(i,"\t:",count)
```

```
Enter a string:emmanuel
e       : 2
m       : 2
a       : 1
n       : 1
u       : 1
l       : 1
```

CO2 Q -26

Add ‘ing’ at the end of a given string.If it already ends with ‘ing’, then add ‘ly’

Program code:

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

Output:

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

```
Enter a string:work
working
```

CO2 Q -27

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

Program code:

```
lis=[]
n=int(SOURCE CODE("Enter the range:"))
print("Enter the words:")
for i in range(0,n):
 lis.append(SOURCE CODE(""))
longest=lis[0]
for i in range(1,n):
 if(len(lis[i])>len(longest)):
 longest=lis[i]
print("Length of longest word is",len(longest))
```

Output:

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

```
Enter the range:5
Enter the words:
Hello
World
How
Are
You
Length of longest word is 5
```

CO2 Q -28

Construct following patterns using nested loop

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

Program Code:

```
for i in range(1,6):
 for j in range(1,i+1):
 print("*",end=" ")
 print("\n")
```

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

Output:

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

```
*

* *

* * *

* * * *

* * * * *

* * * *

* * *

* *

*
```

CO2 Q -29

Generate all factors of a number.

Program code:

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

Output:

COURSE OUTCOME 3

CO3 Q -30

Work with built-in packages

Create a package graphics with modules rectangle, circle and sub package 3D (td) - graphics with modules cuboid & 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).

Program code:

Circle.py

```
from math import pi
def area_circle(radius):
 return pi*radius*radius
def perimeter_circle(radius):
 return 2*pi*radius
```

rectangle.py

```
def area_rec(length,width):
 return length*width
def perimeter_rec(length,width):
 return 2*(length+width)
```

sphere.py

```
from math import pi
def area_sphere(radius):
 return 4*(pi*radius*radius)
def perimeter_sphere(radius):
 return 2*pi*radius
```

cuboid.py

```
def area_cuboid(l,b,h):
 return 2*(l*h + b*h + l*b)
def volume_cuboid(l,b,h):
 return l*b*h
```

driver1.py

```
import Graphics1
from Graphics1 import circle,rectangle
from Graphics1.tdgraphics import cuboid,sphere
from Graphics1.circle import *
print("Area of a circle with radius 10 is : ",circle.area_circle(10))
print("Permeter of a circle with radius 10 is ",circle.perimeter_circle(10))
print("\n")
print("Area of a Rectangle with length and width 10 is : ",rectangle.area_rec(10,10))
print("Permeter of a Rectangle with length and width 10 is :
",rectangle.perimeter_rec(10,10))
print("\n")
```

```
print("Area of a cuboid with length,width,height 10 is :
",cuboid.area_cuboid(10,10,10))
print("Volume of a cuboid with length,width,height 10 is :
",cuboid.volume_cuboid(10,10,10))
print("\n")
print("Area of a spere with radius 10 is : ",sphere.area_sphere(10))
print("Permeter of a spere with radius 10 is ",sphere.perimeter_sphere(10))
```

Output:

```
Command Prompt
Microsoft Windows [Version 10.0.19044.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS>cd Desktop

C:\Users\ASUS\Desktop>cd python

C:\Users\ASUS\Desktop\python>md Graphics1

C:\Users\ASUS\Desktop\python>cd graphics1

C:\Users\ASUS\Desktop\python\Graphics1>notepad circle.py

C:\Users\ASUS\Desktop\python\Graphics1>notepad rectangle.py

C:\Users\ASUS\Desktop\python\Graphics1>md tdgraphics

C:\Users\ASUS\Desktop\python\Graphics1>cd tdgraphics

C:\Users\ASUS\Desktop\python\Graphics1\tdgraphics>notepad cuboid.py

C:\Users\ASUS\Desktop\python\Graphics1\tdgraphics>notepad sphere.py

C:\Users\ASUS\Desktop\python\Graphics1\tdgraphics>cd..

C:\Users\ASUS\Desktop\python\Graphics1>cd..
```

```
C:\Users\ASUS\Desktop\python>notepad driver1.py

C:\Users\ASUS\Desktop\python>python driver1.py
Area of a circle with radius 10 is :  314.1592653589793
Permeter of a circle with radius 10 is  62.83185307179586


Area of a Rectangle with length and width 10 is :  100
Permeter of a Rectangle with length and width 10 is :  40


Area of a cuboid with length,width,height 10 is :  600
Volume of a cuboid with length,width,height 10 is :  1000


Area of a spere with radius 10 is :  1256.6370614359173
Permeter of a spere with radius 10 is  62.83185307179586

C:\Users\ASUS\Desktop\python>
```

COURSE OUTCOME 4

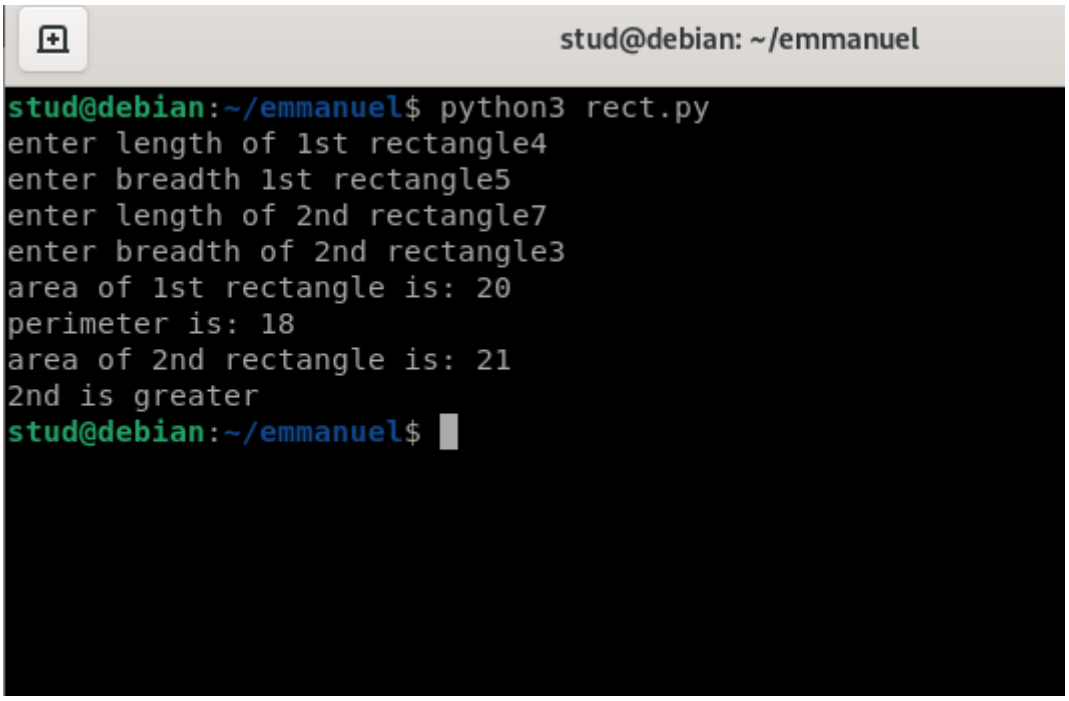
CO4 Q -31

Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.

Program code:

```
class Rectangle:
def __init__(self,length,breadth):
self.length  = length
self.breadth  = breadth
def area(self):
return self.length  * self.breadth
def perimeter(self):
return 2*(self.length  + self.breadth)
c=int(input("enter  length  of 1st rectangle"))
d=int(input("enter  breadth 1st rectangle"))
u=int(input("enter  length  of 2nd rectangle"))
v=int(input("enter  breadth of 2nd rectangle"))
r1= Rectangle(c,d)
r3= Rectangle(u,v)
a=r1.area()
b=r3.area()
print("area  of 1st rectangle  is:",a)
print("perimeter  is:",r1.perimeter())
print("area  of 2nd rectangle  is:",b)
if (a>b):
print("1st  is  greater")
else:
        print("2nd  is  greater")
```

Output:



CO4 Q -32

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.

Program code:

```
class Bank:
def __init__(self,acc_no,name,type_of_acc,balance):
self.acc_no= acc_no
self.name=name
self.type_of_acc=type_of_acc
self.balance=balance
def deposit(self,x):
self.balance=self.balance+x
print("balance  after deposit is=",self.balance)
def withdraw(self,y):
self.balance=self.balance-y
print("balance  after withdrawal  is=",self.balance)
x=int(input("amount  to be deposited"))
y=int(input("amount  to withdraw"))
ob1=Bank(1,"aaa","ccc",300000)
ob2=Bank(2,"bbb","ccc",500000)
ob1.deposit(x)
ob1.withdraw(y)
ob2.deposit(x)
ob2.withdraw(y)
```

Output:

```
amount to be deposited20000
amount to withdraw30000
balance after deposit is= 320000
balance after withdrawal is= 290000
balance after deposit is= 520000
balance after withdrawal is= 490000
```

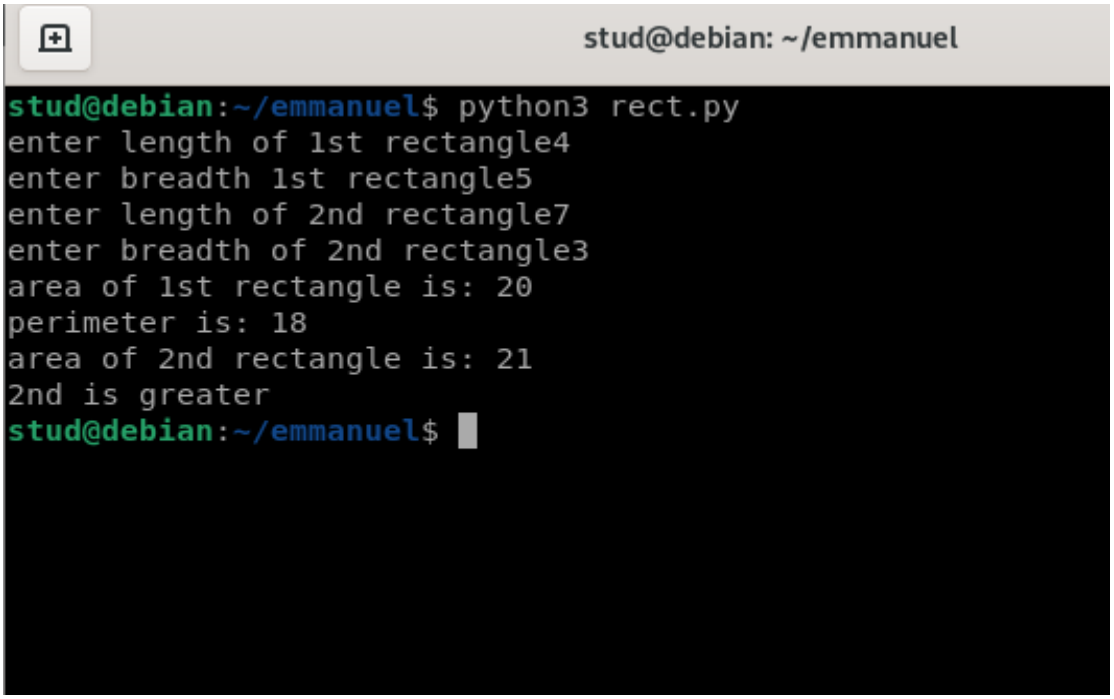
CO4 Q -33

Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.

Program code:

```
class Rectangle:
def __init__(self,length,breadth):
self.length  = length
self.breadth  = breadth
def area(self):
return self.length  * self.breadth
def perimeter(self):
return 2*(self.length  + self.breadth)
def __lt__(self,rr):
if (self.length  >rr .length  and self.breadth  > rr.breadth):
print("Area  of first rectangle  is greater")
else:
        print("Area  of second rectangle  is greater")
c=int(input("enter  length  of 1st rectangle"))
d=int(input("enter  breadth 1st rectangle"))
u=int(input("enter  length  of 2nd rectangle"))
v=int(input("enter  breadth of 2nd rectangle"))
r1= Rectangle(c,d)
r3= Rectangle(u,v)
a=r1.area()
b=r3.area()
print("area  of 1st rectangle  is:",a)
print("perimeter  is:",r1.perimeter())
print("area  of 2nd rectangle  is:",b)
        r1 < r3
```

Output:



CO4 Q -34

Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time.

Program code:

```
class Time:
def __init__(self,hr,min,sec):
self.hr=hr
self.min=min
self.sec=sec
def __add__ (self,t):
return(self.hr+t.hr,self.min+t.min,self.sec+t.sec)
t1=Time(3,20,35)
t2=Time(2,25,40)
        print(t1+t2)
```

Output:

```
(5, 45, 75)
```

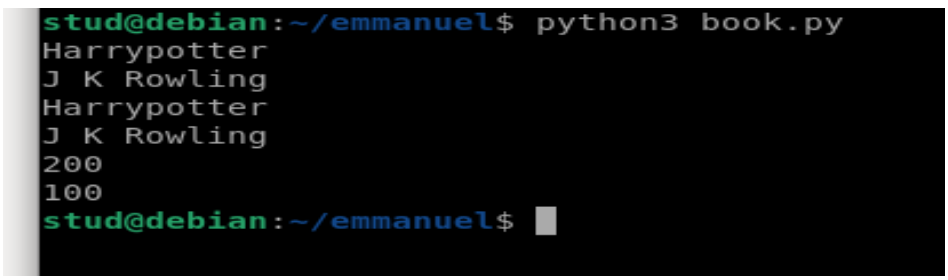
CO4 Q -35

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.

Program code:

```
class Publisher(object):
        def __init__(self,name):
                self.name=name
        def display1(self):
                print(self.title)
                print(self.author)
class Book(Publisher):
        def __init__(self,name,title,author):
                super().__init__(name)
                self.title=title
                self.author=author
        def display2(self):
                super().display1()
                print(self.title)
                print(self.author)
class Python(Book):
        def __init__(self,name,title,author,price,no_of_pages):
                super().__init__(name,title,author)
                self.price=price
                self.no_of_pages=no_of_pages
        def display3(self):
                super().display2()
                print(self.price)
                print(self.no_of_pages)
p=Python("abc Publications","Harrypotter","J  K Rowling",200,100)
p.display3()
```

Output:



COURSE OUTCOME 5

CO5 Q -36

Write a python program to read a file line by line and store it into a list.

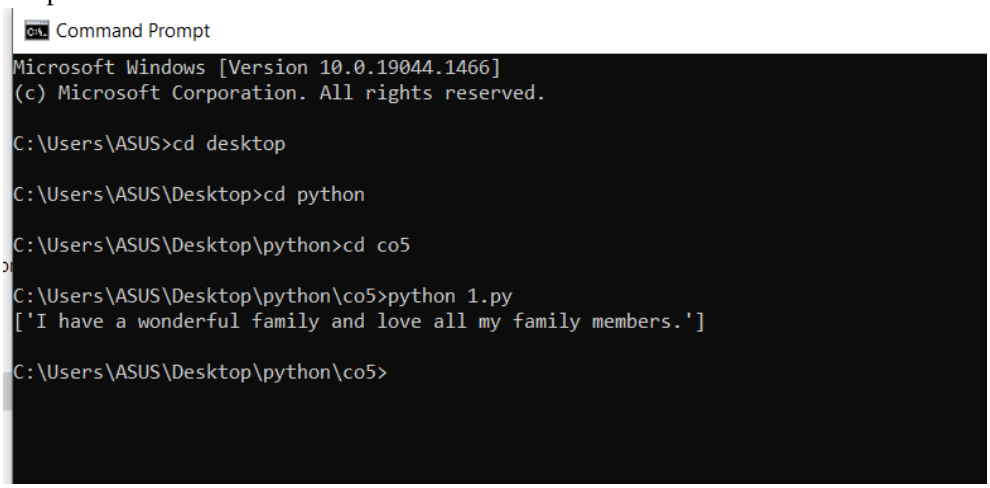
Program code:

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

text.txt

```
I have a wonderful family and love all my family members.
```

Output:



CO5 Q -37

Write a python program to read each row from a given csv file and print a list of strings

Program code:

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

work.csv

	A	B	C	D
1	Name	Age	Profession	
2	Das	40	Manager	
3	Vinu	38	Ass.Manager	
4	Manu	35	Staff	
5	Janaki	30	Nurse	
6				

Output: