# ASSIGNMENT

# **Lab-3 Experiments**

Anu Priya

•

3:05 PM

Due 5:30 PM

Note: Execute the list of experiments attached and upload the screenshot of the result with its source code within the time limit. Properly execute all the experiments with its logic, as I may ask for implementation logic, and it's working in viva voce.

PREPARED BY:

SAMAR MITTAL

TASK 1:

str=input("Enter string : ")

a=dict()

for ch in str:

    if ch not in a:

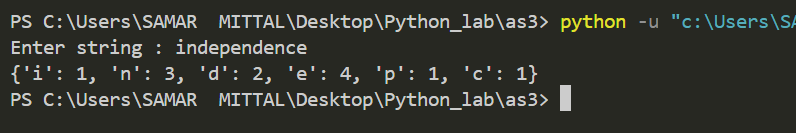
        a[ch]=1

    else:

        a[ch]=a[ch]+1

print(a)

output:



TASK 2:

name1=input("Enter name ")

a=input("Enter birthday in dd-mm-yy : ")

b=a.split("-")

name2=input("Enter name ")

x=input("Enter birthday in dd-mm-yy : ")

y=x.split("-")

ram={"Day":b[0],"Month":b[1],"Year":b[2]}

shyam={"Day":y[0],"Month":y[1],"Year":y[2]}

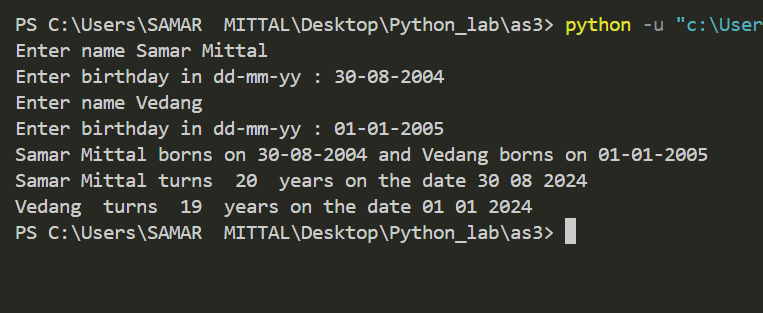
res1=name1+" borns on " + a +" and "+name2+" borns on "+x

print(res1)

print(name1,"turns ",2024-int(ram["Year"])," years on the date",ram["Day"],ram["Month"],2024)

print(name2," turns ",2024-int(shyam["Year"])," years on the date",shyam["Day"],shyam["Month"],2024)

OUTPUT:



TASK 3:

*def* gcd(*a*,*b*):

     if (*b*==0):

          return *a*

     return gcd(*b*,*a*%*b*)

*def* lcm(*x*,*y*):

     gc=gcd(a,b)

     return (a\*b)/gc

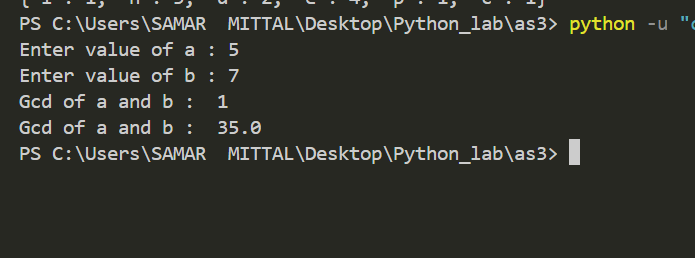
a=int(input("Enter value of a : "))

b=int(input("Enter value of b : "))

print("Gcd of a and b : ",gcd(a,b))

print("Gcd of a and b : ",lcm(a,b))

OUTPUT:



TASK 4:

import math

*def* iscollide(*a*,*b*):

    dis=math.sqrt(pow(*a*[1]-*b*[1],2)+pow(*a*[0]-*b*[0],2))

    rad\_sum=*a*[2]+*b*[2]

    if(dis<=rad\_sum):

        return True

    else:

        return False

a=(0,0,2)

b=(2,0,2)

print("A : ",a,"\nB : ",b)

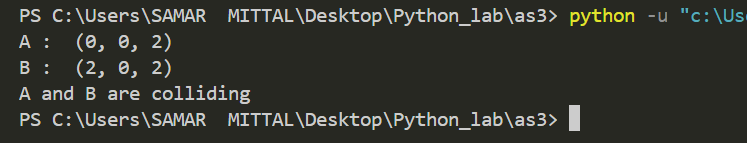
if (iscollide(a,b)):

    print("A and B are colliding")

else:

    print("A and B are not collding")

OUTPUT:



TASK 5:

import statistics

n=int(input("Enter length of list : "))

a=list()

print("Enter list of numbers ")

count=0

for i in range(0,n):

    val=int(input())

    a.insert(i,val)

    count+=val

print("Entered data set: ")

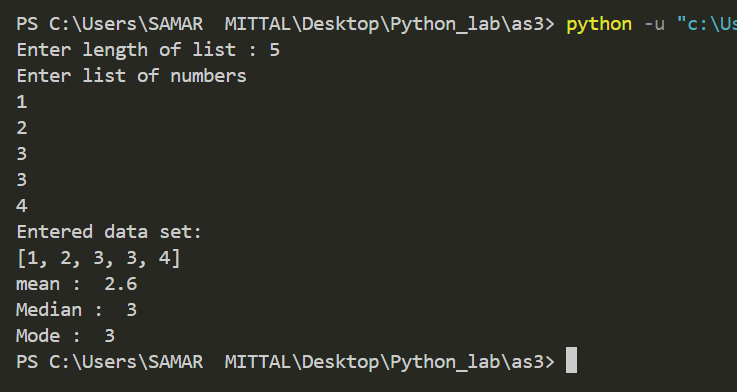
print(a)

print("mean : ",count/5)

print("Median : ",statistics.median(a))

print("Mode : ",statistics.mode(a))

Output:



TASK 6:

*def* selectionsort(*arr*,*n*):

    for i in range(0,*n*):

        min=i

        for j in range(i+1,*n*):

            if(*arr*[j]<*arr*[min]):

                min=j

*arr*[min],*arr*[i]=*arr*[i],*arr*[min]

*def* insertionsort(*arr*,*n*):

    if *n* <= 1:

        return

    for i in range(1, *n*):

        key = *arr*[i]

        j = i-1

        while j >= 0 and key < *arr*[j]:

*arr*[j+1] = *arr*[j]

            j -= 1

*arr*[j+1] = key

*def* bubblesort (*list*,*n*):

    for i in range(*n*-1,0,-1):

        for j in range(0,i):

            if *list*[j]>*list*[i]:

*list*[i],*list*[j]=*list*[j],*list*[i]

*def* merge(*arr*, *l*, *m*, *r*):

    n1 = *m* - *l* + 1

    n2 = *r* - *m*

    L = [0] \* (n1)

    R = [0] \* (n2)

    for i in range(0, n1):

        L[i] = *arr*[*l* + i]

    for j in range(0, n2):

        R[j] = *arr*[*m* + 1 + j]

    i = 0

    j = 0

    k = *l*

    while i < n1 and j < n2:

        if L[i] <= R[j]:

*arr*[k] = L[i]

            i += 1

        else:

*arr*[k] = R[j]

            j += 1

        k += 1

    while i < n1:

*arr*[k] = L[i]

        i += 1

        k += 1

    while j < n2:

*arr*[k] = R[j]

        j += 1

        k += 1

*def* mergesort(*arr*, *l*, *r*):

    if *l* < *r*:

        m = *l*+(*r*-*l*)//2

        mergesort(*arr*, *l*, m)

        mergesort(*arr*, m+1, *r*)

        merge(*arr*, *l*, m, *r*)

*def* sort\_list(*arr*):

    ch=int(input("Enter 1->Bubble Sort\n2->Selection Sort\n3->Merge Sort\n4->Insertion Sort\n"))

    if (ch==1):

        bubblesort(*arr*,len(*arr*))

    elif ch==2:

        selectionsort(*arr*,len(*arr*))

    elif ch==3:

        mergesort(*arr*,0,len(*arr*)-1)

    elif ch==4:

        insertionsort(*arr*,len(*arr*))

    else:

        print("Wrong Entry || exit")

        return

a=[9,7,5,4,2,1]

print("Initial Array : ",a)

sort\_list(a)

print("Sorted Array",a)

OUTPUT:

