Name – Samarth Udagi

MIS – 112315160

Lab 3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q.1. Write a program to count the numbers of characters in the given string and store them in a

dictionary data structure.

Code :

s=input("Enter the string : ")

dict={}

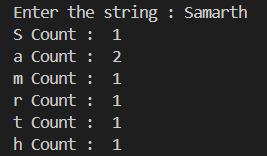
for c in s:

    if c in dict : dict[c]+=1

    else : dict[c]=1

for i in dict : print(i,"Count : ",dict[i])

Output :



Q.2. Write a program to use split and join methods in the given string and trace a birthday with a

dictionary data structure.

Code : rama=input("Enter the birthdate: ")

shyam=input("Enter the birthdate: ")

components1=rama.split("-")

components2=shyam.split("-")

Rama={

"Year":components1[2],

"Month":components1[1],

"Date" :components1[0]

}

Shyam={

"Year":components2[2],

"Month":components2[1],

"Date" :components2[0]

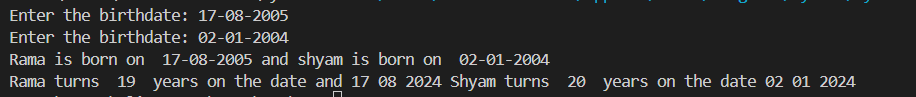
}

print("Rama is born on ",rama,"and shyam is born on ",shyam)

print("Rama turns ",2024-int(Rama["Year"])," years on the date and",Rama["Date"],Rama["Month"],2024,end=" ")

print("Shyam turns ",2024-int(Shyam["Year"])," years on the date",Shyam["Date"],Shyam["Month"],2024)

Output :



Q.3. Write function to compute gcd and lcm of two numbers.

Code :

def gcd(a,b) :

    if b==0 : return a

    return gcd(b,a%b)

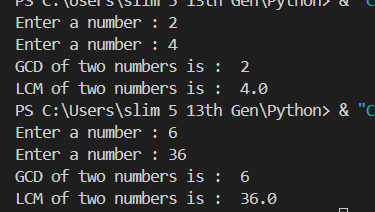
a=int(input("Enter a number : "))

b=int(input("Enter a number : "))

print("GCD of two numbers is : ",gcd(a,b))

print("LCM of two numbers is : ",a\*b/gcd(a,b))

Output :

  
Q.4. Write a function ball \_collide that takes two balls as parameters and computes if they are

colliding. Your function should return a Boolean representing whether or not the balls are

colliding. Represent a ball on a plane as a tuple of (x, y, r), r being the radius. If (distance

between two balls centers) &lt;= (sum of their radii) then (they are colliding)

Code :

import math

def ball\_collide(ball1, ball2):

    x1, y1, r1 = ball1

    x2, y2, r2 = ball2

    distance = math.sqrt((x2 - x1) \*\* 2 + (y2 - y1) \*\* 2)

    return distance <= (r1 + r2)

x1 = float(input("Enter the x-coordinate of the first ball: "))

y1 = float(input("Enter the y-coordinate of the first ball: "))

r1 = float(input("Enter the radius of the first ball: "))

ball1 = (x1, y1, r1)

x2 = float(input("Enter the x-coordinate of the second ball: "))

y2 = float(input("Enter the y-coordinate of the second ball: "))

r2 = float(input("Enter the radius of the second ball: "))

ball2 = (x2, y2, r2)

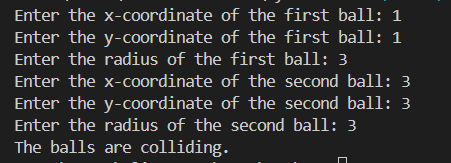
if ball\_collide(ball1, ball2):

    print("The balls are colliding.")

else:

    print("The balls are not colliding.")

Output :



Q.5. Find mean, median, mode for the given set of numbers in a list.

Code :

import statistics

arr = []

n = int(input("Enter number of elements in list : "))

print("Enter elements : ")

sum = 0

for i in range(0, n):

    ele = int(input())

    arr.append(ele)

    sum+=ele

print("Given list is : ")

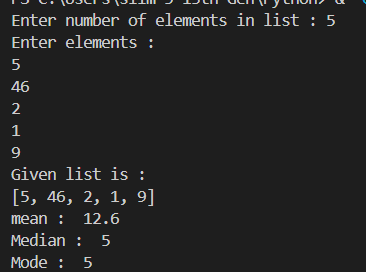
print(arr)

print("mean : ",sum/n)

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

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

Output :

  
Q.6. Write a program to implement

a. Bubble sort,

b. Merge sort,

c. Selection sort and

d. Insertion sort.

Execute these sorting algorithms using switch case.

Code :

def bubble(arr):

    n = len(arr)

    for i in range(n):

        for j in range(0, n-i-1):

            if arr[j] > arr[j+1]:

                arr[j], arr[j+1] = arr[j+1], arr[j]

def merge(arr):

    if len(arr) > 1:

        mid = len(arr) // 2

        L = arr[:mid]

        R = arr[mid:]

        merge(L)

        merge(R)

        i = j = k = 0

        while i < len(L) and j < len(R):

            if L[i] < R[j]:

                arr[k] = L[i]

                i += 1

            else:

                arr[k] = R[j]

                j += 1

            k += 1

        while i < len(L):

            arr[k] = L[i]

            i += 1

            k += 1

        while j < len(R):

            arr[k] = R[j]

            j += 1

            k += 1

def selection(arr):

    for i in range(len(arr)):

        min\_idx = i

        for j in range(i+1, len(arr)):

            if arr[j] < arr[min\_idx]:

                min\_idx = j

        arr[i], arr[min\_idx] = arr[min\_idx], arr[i]

def insertion(arr):

    for i in range(1, len(arr)):

        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 sort\_array(algorithm, arr):

    algorithms = {

        'bubble': bubble,

        'merge': merge,

        'selection': selection,

        'insertion': insertion

    }

    if algorithm in algorithms:

        algorithms[algorithm](arr)

    else:

        print("Invalid algorithm choice")

n=int(input("Enter the size of array : "))

arr = []

print("Enter the elements of array ")

for i in range(0,n):

    x=int(input())

    arr.append(x)

algorithm\_choice = input("Enter the sorting algorithm (bubble, merge, selection, insertion): ").strip().lower()

arr\_copy = arr.copy()

sort\_array(algorithm\_choice, arr\_copy)

print(f"Sorted array using {algorithm\_choice} sort: {arr\_copy}")

Output :   
