# **Name: Abdurrahman Qureshi**

# **Roll No: 242466**

Practical No: 3

**1) To check whether a given no is prime or not.**

CODE:

def prime(x):

    isPrime = True

    for i in range(2,x):

        if x % i == 0:

            isPrime = False

            break

        else:

            isPrime = True

    return isPrime

x = int(input("Enter an integer:"))

isPrime = prime(x)

if isPrime:

    print(x, "is a prime number.")

else:

    print(x, "is not a prime number.")

OUTPUT:



**2) To create a Fibonacci series of n terms.**

CODE:

def Fibonacci(x):

    y = 0

    z = 1

    print(y ,end=" ")

    print(z, end=" ")

    for i in range(0,x):

        a = y + z

        print(a, end=" ")

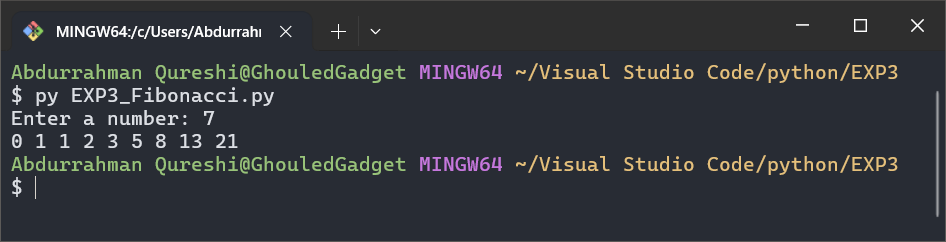
        y = z

        z = a

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

Fibonacci(x)

OUTPUT:



**3) To create 5 different list using list comprehension method**

CODE:

list\_a = [x for x in range(1,11)]

print(list\_a)

list\_b = [x for x in range(1,11,2)]

print(list\_b)

list\_c = [x for x in range(0,11,2)]

print(list\_c)

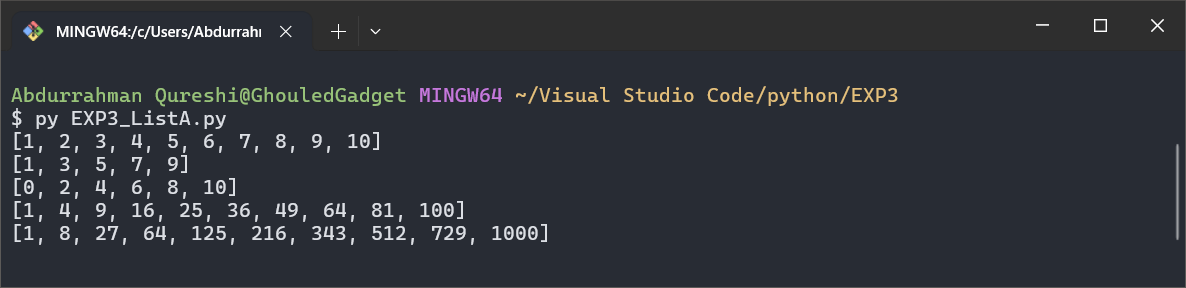
list\_d = [x\*\*2 for x in range(1,11)]

print(list\_d)

list\_e = [x\*\*3 for x in range(1,11)]

print(list\_e)

OUTPUT:



**4) Create list, add random 25 numbers in list and print largest number of list [ hint : use ( print( max( L),min, reversed and sum-)**

CODE:

import random

List\_random = []

for i in range(0,26):

x = random.randint(1,100)

List\_random.append(x)

print(List\_random)

print("Original List\_random: " , List\_random)

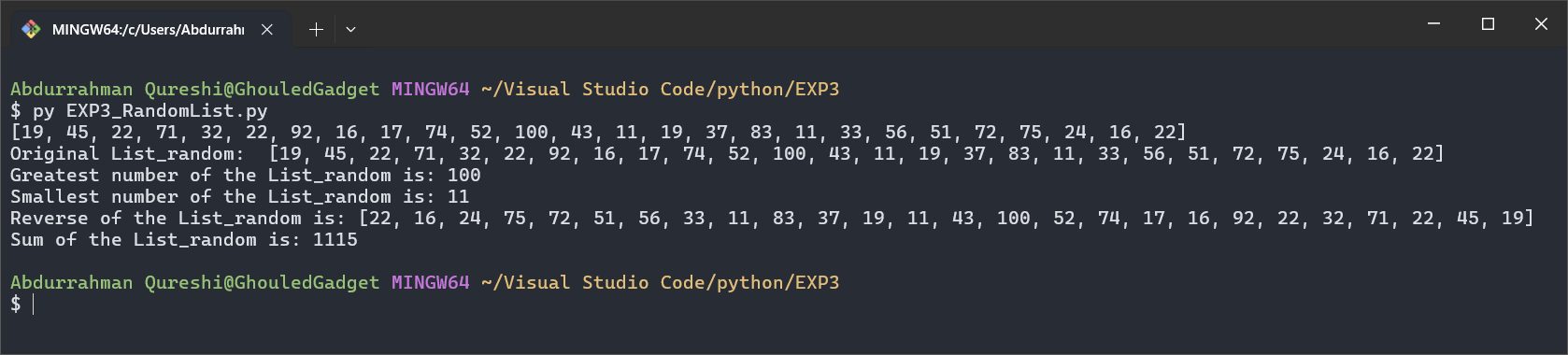
print("Greatest number of the List\_random is:" , max(List\_random))

print("Smallest number of the List\_random is:" , min(List\_random))

print("Reverse of the List\_random is:", list(reversed(List\_random)))

print("Sum of the List\_random is:" , sum(List\_random))

OUTPUT:



**5) To check whether a given number is present in the list or not? first accept elements in the list .**

CODE:

list = []

list\_len = int(input("Enter list length: "))

for i in range(0,list\_len):

x = int(input("Enter an integer: "))

list.append(x)

print("Original List:", list)

to\_search = int(input("Enter number to search: "))

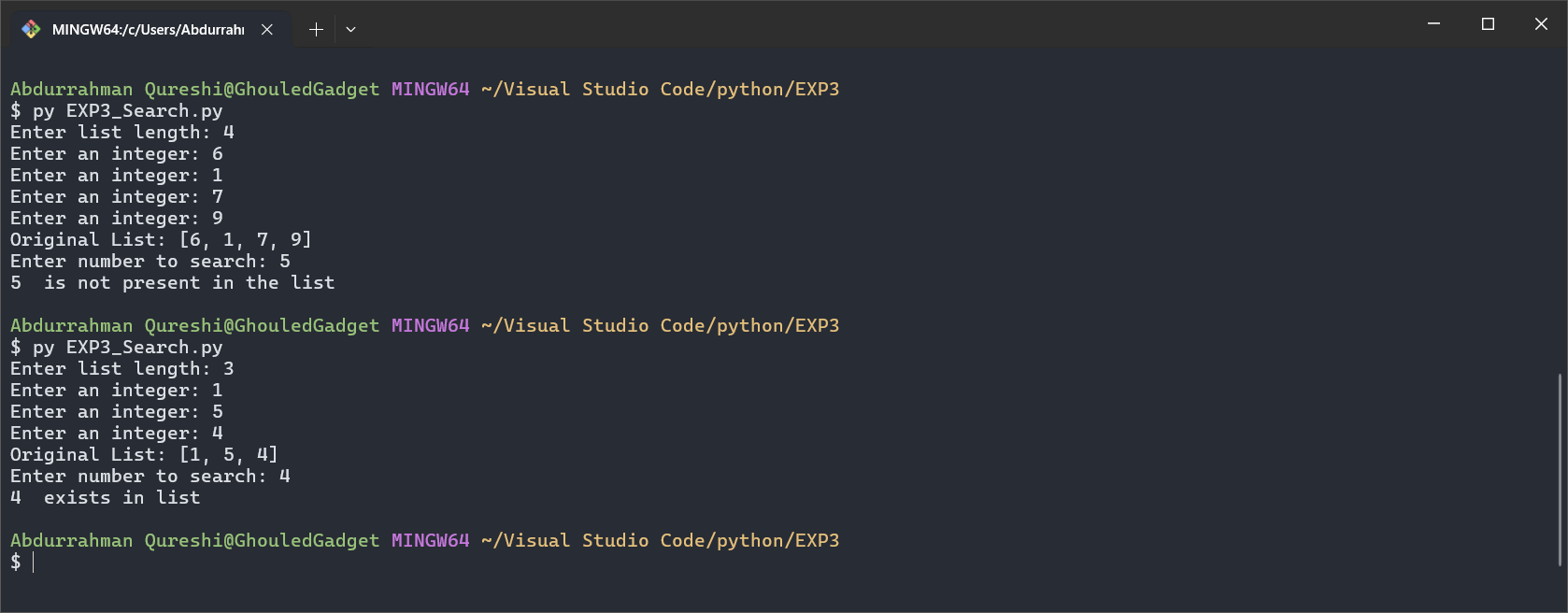
if to\_search in list:

print(to\_search , " exists in list")

else:

print(to\_search , " is not present in the list")

OUTPUT:



**6) To demonstrate any 10 methods of List [ put comments in code and explain the method ]**

CODE:

x = [7, 91, 13, 34, 8, 5, 93, 37, 80, 83, 27, 61, 30, 78, 94, 19, 68, 55, 63, 9, 29, 70, 69, 9, 52]

print(x) *#* print the list

print("Max element: " , max(x)) *#* max element

print("Min element: " , min(x)) *#* min element

print("Sum of list elements: " ,sum(x)) *#* sum of list elements

print("Length of list elements: " , len(x)) *#* length of list elements

print("Appending 34:" , x.append(34)) *#* appends 34

print("Popping from list: " , x.pop(0)) *#* pops the first element

print("Sorted list elements: " , sorted(x)) *#* sorts the list elements

print("Counting  '9' in list: " , x.count(9)) *#* counts the number of 9s

print("Inserting '76' at '4' index: " , x.insert(1,76)) *#* inserts '76' at index '4'

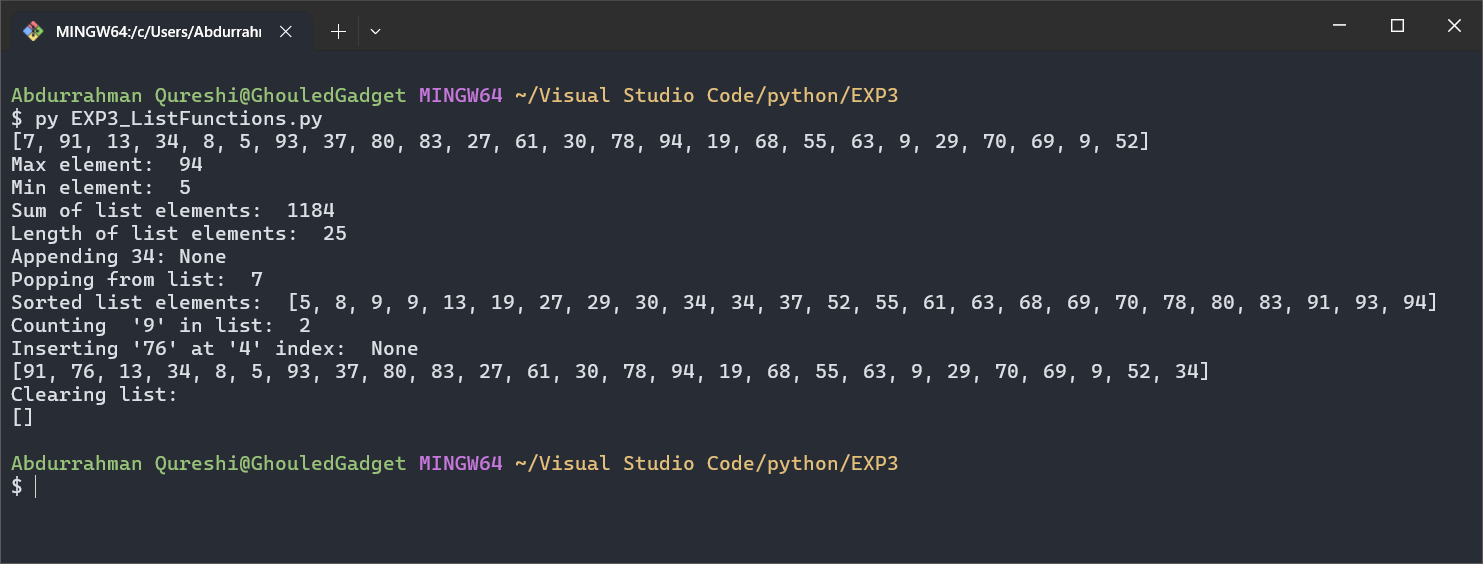
print(x)*#* print the list

print("Clearing list:") *#* clears the list

x.clear()

print(x)*#* print the list

OUTPUT:



**7) To add two list [ adding elements of list ] and concatenate two list  (Exmaple below L3 is adding element L4 is concatenation )L1=[1,2,3] ,L2=[11,12,13] L3=[12,14,16] , L4=[1,2,3,11,12,13]**

CODE:

l1 = [7, 91, 13, 34]

l2 = [8, 5, 93, 37]

l3 = l1 + l2

l4 = []

for i in l1:

    for j in l2:

        l4.append(i + j)

        break

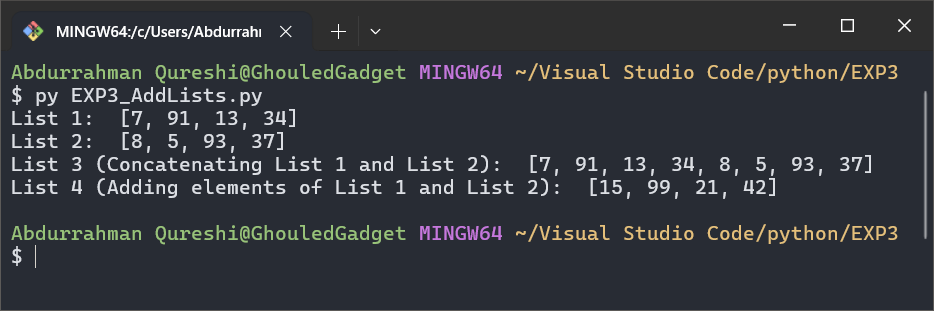
print("List 1: ",l1)

print("List 2: ",l2)

print("List 3 (Concatenating List 1 and List 2): ",l3)

print("List 4 (Adding elements of List 1 and List 2): ",l4)

OUTPUT:



**8) To implement a stack using a list . Define push and pop functions**

CODE:

stack = []

def switch(choice):

    if choice == 1:

        item = input("Enter an item to push: ")

        stack.insert(0, item)

    elif choice == 2:

        if len(stack) == 0:

            print("Stack is empty. Cannot pop.")

        else:

            first = stack[0]

            print("Popped item is:", stack.remove(first))

            print(stack)

    elif choice == 3:

        if len(stack) == 0:

            print("Stack is empty.")

        else:

            print("Stack:", stack)

    else:

        print("Invalid choice.")

choice = 1

while (choice != 0):

    print("\n1. Push")

    print("2. Pop")

    print("3. Display")

    print("0. Exit")

    x = int(input("Enter your choice: "))

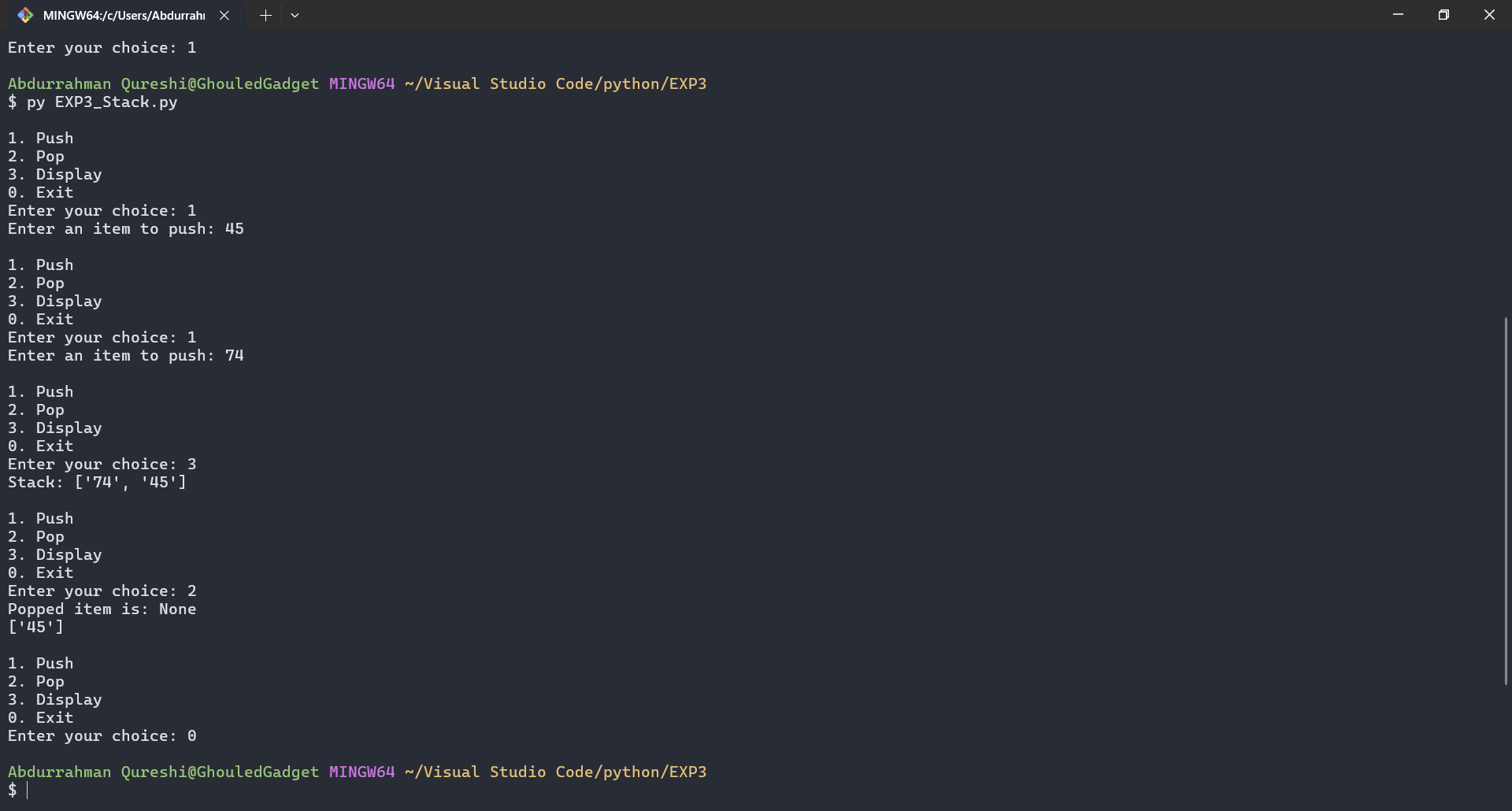
    if (x == 1):

        break

    else :

        switch(x)

OUTPUT:



9) To implement a game [Rock , Paper , Scissor ] using a list . the game should keep on executing as long as player says ‘y’ [ Hint ]

while(ch==’y’):

        ---

----

Print ( “ do you wish to continue to play game ? (y/n)”)

 ch=input( )

CODE:

import random

options = ["ROCK", "PAPER", "SCISSOR"]

score = []

choice = "Y"

def round\_winner(player, computer):

    if player == computer:

        return "TIE"

    elif (player == "ROCK" and computer == "SCISSOR") or (player == "SCISSOR" and computer == "PAPER") or (player == "PAPER" and computer == "ROCK"):

        return "PLAYER"

    else:

        return "COMPUTER"

while (choice.upper() != "N"):

    computer\_choice = options[random.randint(0, 2)]

    player\_choice = input("Enter your choice (rock, paper, scissor): ").upper()

    if player\_choice not in options:

        print("Invalid choice! Please enter ROCK, PAPER, or SCISSOR.")

        continue

    result = round\_winner(player\_choice, computer\_choice)

    print("\nPlayer chose:", player\_choice)

    print("Computer chose:", computer\_choice)

    print("Round Winner:", result, "\n")

    score.append(result)

    choice = input("Do you wish to continue to play the game? (y/n): ").upper()

    if choice not in ["Y", "N"]:

        print("Invalid input, please enter 'y' or 'n'.")

    elif choice == "N":

        print("\nGame Over!")

        print("Final Score:")

        print("Player: ", score.count("PLAYER"))

        print("Computer: ", score.count("COMPUTER"))

        print("Ties: ", score.count("TIE"))

OUTPUT:



**10) To sort a given number list in ascending order using functions .take numbers from user**

 def sortList(L):

………

……..

L=[]

sortList(L)

CODE:

def sortList(L):

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

    for i in range(n):

        L.append(int(input("Enter an integer for position [" + str(i) + "]: ")))

    print("Original List:", L)

    print("\nVia sort():")

    L\_sort = sorted(L)

    print(L\_sort)

    print("\nVia Bubble Sort:")

    L\_bubble = L

    for i in range(len(L\_bubble)):

        for j in range(len(L\_bubble) - 1):

            if L\_bubble[j] > L\_bubble[j + 1]:

                temp = L\_bubble[j]

                L\_bubble[j] = L\_bubble[j + 1]

                L\_bubble[j + 1] = temp

    print(L\_bubble)

L = []

sortList(L)

OUTPUT:

