**Programming Assignment 14**

Q.1) Define a class with a generator which can iterate the numbers, which are divisible by 7, between a given range 0 and n.

Ans.)

class Task1:

    def gen(self, n):

        for i in range(n+1):

            if i%7 == 0:

                yield i

nums = Task1()

[i for i in nums.gen(100)]

Q.2) Write a program to compute the frequency of the words from the input. The output should output after sorting the key alphanumerically.

Ans.)

def task2(string: str):

    string\_list = string.split(' ')

    out\_dict = {}

    for s in string\_list:

        if s in out\_dict.keys():

            out\_dict[s] += 1

        else:

            out\_dict[s] = 1

    for k, v in sorted(out\_dict.items()):

        print(k, ':', v)

Q.3) Define a class Person and its two child classes: Male and Female. All classes have a method "getGender" which can print "Male" for Male class and "Female" for Female class.

Ans.)

class Person:

    def getGender(self):

        pass

class Male(Person):

    def getGender(self):

        print('Male')

class Female(Person):

    def getGender(self):

        print('Female')

male = Male()

female= Female()

male.getGender()

female.getGender()

Q.4) Please write a program to generate all sentences where subject is in ["I", "You"] and verb is in ['Play', "Love"] and the object is in ["Hockey","Football"].

Ans.)

subject=["I", "You"]

verb=["Play", "Love"]

obj=["Hockey","Football"]

sentence\_list = []

for i in subject:

    for j in verb:

        for k in obj:

            sentence\_list.append(i + " " + j + " " + k)

for sentence in sentence\_list:

    print(sentence)

Q.5) Please write a program to compress and decompress the string "hello world!hello world!hello world!hello world!"

Ans.)

import zlib

string = 'hello world!hello world!hello world!hello world!'

compress\_string = zlib.compress(bytes(s, 'utf-8'))

print(f"Compressed string is: {compress\_string}")

print(f"Decompressed string is: {zlib.decompress(compress\_string)}")

Q.6) Please write a binary search function which searches an item in a sorted list. The function should return the index of element to be searched in the list.

Ans.)

def binarySearch(list1, n):

    low = 0

    high = len(list1) - 1

    mid = 0

    while low <= high:

        mid = (high + low) // 2

        if list1[mid] < n:

            low = mid + 1

        elif list1[mid] > n:

            high = mid - 1

        else:

            return mid

    return -1