1. \*\*Find the leap year :

y=int(input("enter the anniversary year:"))

if y % 4==0:

print("anniversary year is leap year")

elif y%100 == 0:

print("Not a leap year")

elif y%400 == 0:

print("leap year")

x = y% 4

if y % 4 ==0:

print("nxt anniversary",y+4)

if y % 4 !=0:

print("previous year",y-x)

2.\*\*non prime number between x and y:

x=int(input("enter x:"))

y=int(input("enter y:"))

for i in range(x,y+1):

if i>1:

for n in range(2,i):

if i%n==0:

print(i)

break

3.\*\*Pythagorean triples between limit:

limit=int(input("enter the limit"))

triples=[(a,b,c) for a in range(1,limit) for b in range(1,limit) for c in range(1,limit) if a\*\*2+b\*\*2==c\*\*2]

print("pythagorean triples within the limit")

for triple in triples:

print(triple)

4.\*\*find the lcm and gcd of given numbers:

n1=int(input("enter the number:"))

n2=int(input("enter the number:"))

t1=n1

t2=n2

while n1%n2!=0:

r=n1%n2

n1=n2

n2=r

gcd=n2

print("gcd:",gcd)

lcm=n1\*n2//gcd

print("lcm:",lcm)

5.\*\* tanspose of matrix:

a=[[1,2,3],

[4,5,6],

[7,8,9]]

rows=len(a)

cols=len(a[0])

t=[[0,0,0],

[0,0,0],

[0,0,0]]

for i in range(0,cols):

for j in range(0,rows):

t[i][j]=a[j][i]

print("transpose of matrix")

for i in range(0,cols):

for j in range(0,rows):

print(t[i][j],end=" ")

print(" ")

6.\*\*sum of row , columns and diagonals in matrix:

**a=[[1,2,3],**

**[4,5,6],**

**[7,8,9]]**

**rows=len(a)**

**cols=len(a[0])**

**diagonal=len**

**#sum of rows**

**for i in range(0,rows):**

**rowsum=0**

**for j in range(0,cols):**

**rowsum=rowsum+a[i][j]**

**print("sum of " +str(i+1)+"rows:"+str(rowsum))**

**#sum of columns**

**for i in range(0,rows):**

**colssum=0**

**for j in range(0,cols):**

**colssum=colssum+a[j][i]**

**print("sum of " +str(i+1)+"columns:"+str(colssum))**

**#diagonal sum**

**diagonalsum=0**

**for k in range(0,rows):**

**diagonalsum=diagonalsum+a[k][k]**

**print("diagonal sum",diagonalsum)**

**7.\*\*multiplication of matrix**

**a=[[12,7,3],**

**[4,5,6],**

**[7,8,9]]**

**b=[[5,8,1,2],**

**[6,7,3,0],**

**[4,5,9,1]]**

**c=[[0,0,0,0],**

**[0,0,0,0],**

**[0,0,0,0]]**

**for i in range (len(a)):**

**for j in range(len(b[0])):**

**for k in range(len(b)):**

**c[i][j]=c[i][j]+a[i][k]\*b[k][j]**

**for i in c:**

**print(i)**

**8.\*\*find the sum of two matrix:**

**a=[[1,2,3],**

**[3,4,5]]**

**b=[[3,5,6],**

**[6,7,8]]**

**c=[[0,0,0],**

**[0,0,0]]**

**for i in range(len(a)):**

**for j in range(len(a[0])):**

**c[i][j]=a[i][j]+b[i][j]**

**for i in c:**

**print(i)**

**9.\*\*which is greatest binary number**

**a='1101'**

**b='1110'**

**c='1111'**

**bina=int(a,2)**

**binb=int(b,2)**

**binc=int(c,2)**

**if bina>binb and bina>binc:**

**print("a is greatest",a)**

**elif binb>bina and binb>binc:**

**print("b is greatest",b)**

**else:**

**print("c is greatest",c)**

**def max\_profit(prices):**

**total\_profit=0**

**for i in range(1,len(prices)):**

**if prices[i]>prices[i-1]:**

**total\_profit+=prices[i]-prices[i-1]**

**return total\_profit**

**print(max\_profit([7,1,5,3,6,4]))**

**print(max\_profit([7,6,4,3,1]))**

**print(max\_profit([10,22,5,75,65,80]))**

**print(max\_profit([2,30,15,10,8,25,80]))**

**print(max\_pro fit([10,22,5,75,65,80]))**

**10.\*\* sum of two binary numbers:**

**num1=input("enter the binary number1=")**

**num2=input("enter the binary number2=")**

**sum=bin(int(num1,2)+int(num2,2))**

**print("sum of binary numbers",sum[2:])**

**11.\*\*convert binary to decimal ,octal,hexadecimal:**

**num=input("enter the binary number:")**

**dec=int(num,2)**

**oct=oct(dec)**

**hex=hex(dec)**

**print("decimal number",dec)**

**print("octal number",oct[2:])**

**print("hexadecimal",hex[2:])**

**12.\*\*find the square and cube of given number:**

**num=float(input("enter the number:"))**

**sqr=num\*\*2**

**cube=num\*\*3**

**print("square:",sqr)**

**print("cube:",cube)**

**13.\*\*print the unique permutations of given number:**

**num1=int(input("enter the number:"))**

**num2=int(input("enter the number:"))**

**num3=int(input("enter the number:"))**

**list=[]**

**list.append(num1)**

**list.append(num2)**

**list.append(num3)**

**print("list",list)**

**for i in range (0,3):**

**for j in range (0,3):**

**for k in range (0,3):**

**if (i!=j and j!=k and k!=i):**

**print("permutations are:",list[i],list[j],list[k])**

**14.\*\*to print number of factors and print nth factors:**

**num=int(input("enter the number:"))**

**list=[]**

**for i in range(1,num+1):**

**if num%i==0:**

**list.append(i)**

**print(list)**

**length=len(list)**

**print("number of factors:",length)**

**x=int(input("enter the nth factor:"))**

**for i in range(0,x):**

**if x>length:**

**print("invalid")**

**break**

**else:**

**print(list[i],end=" ")**

**15.\*\*Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. She is being offered 15 percent rate of interest; he is being offered 12 percent rate of interest for all other customers, the ROI is 10 percent:**

**p=int(input("enter the principal amount:"))**

**t=int(input("enter the no of years:"))**

**g=input("enter the male/female:")**

**sc=input("enter senior citizen=y/n:")**

**if sc=='y' and g=='f':**

**r=15**

**elif sc=='y' and g=='m':**

**r=12**

**else:**

**r=10**

**si=(p\*t\*r)/100**

**print("sample intreset:",si)**

**16.\*\*cimbling a staircases,it takes n steps to reach the top.each time you can either climb 1 or 2 steps.how many ways can you climb:**

**s=int(input("enter the number:"))**

**a,b=0,1**

**if s<=1:**

**print("number of ways:",s)**

**else:**

**for i in range(2,s+2):**

**fib=b+a**

**a,b=b,b+a**

**print("number of ways:",fib)**

**17.\*\*fabinocii series are:**

**a,b=0,1**

**num=int(input("enter the number:"))**

**print(a)**

**while b<num:**

**print(b)**

**a,b=b,b+a**

**18.\*\* find the sum of digits in triangle:**

**triangle=[**

**[1],**

**[2,3],**

**[4,5,6],**

**[7,8,9,5]]**

**total\_sum=sum(int(digits) for rows in triangle for digits in rows)**

**print("total\_sum",total\_sum)**

**19.\*\* difference between square of numbers and sum of squares:**

**n=int(input("enter the number:"))**

**sum\_of\_squares=0**

**sum\_of\_numbers=0**

**for i in range(1,n+1):**

**sum\_of\_squares=sum\_of\_squares+i\*\*2**

**sum\_of\_numbers=sum\_of\_numbers+i**

**square\_of\_numbers=sum\_of\_numbers\*\*2**

**difference=square\_of\_numbers-sum\_of\_squares**

**print("difference:",difference)**

**20.\*\*find the sum of 1!/1+2!/2+.......N!/N numbers:**

**n=int(input("enter the number:"))**

**fact=1**

**sum=0**

**for i in range(1,n+1):**

**fact=fact\*i**

**sum=sum+fact/i**

**print("sum:",int(sum))**

**21.\*\*find the sum of 1!+2!+3!+……..+n! numbers:**

**n=int(input("enter the number:"))**

**fact=1**

**sum=0**

**for i in range(1,n+1):**

**fact=fact\*i**

**sum=sum+fact**

**print("sum",sum)**

**22.\*\*factorial of given number:**

**n=int(input("enter the number:"))**

**fact=1**

**if n<0:**

**print("factorial does not exits for negative number:")**

**elif n==0:**

**print("factorial of 0 is :1")**

**else:**

**for i in range(1,n+1):**

**fact=fact\*i**

**print("factorial of number:",fact)**

**23.\*\*find the sum of 1^2+2^2+---+n^2 are:**

**n=int(input("enter the number:"))**

**sum=0**

**for i in range(1,n+1):**

**sum=sum+i\*\*2**

**print("sum",sum)**

**24.\*\*find the sum of n numbers:**

**n=int(input("enter the number:"))**

**sum=0**

**for i in range(1,n+1):**

**sum=sum+i**

**print("sum",sum)**

**25.\*\*spiral matrix:**

**a=[[1,2,3],**

**[4,5,6]]**

**while a:**

**print(\*(a.pop(0)),end=" ")**

**a=list(zip(\*a))[::-1]**

**26.\*\*tech number or not:**

**n=int(input("enter the number:"))**

**m=str(n)**

**a=m[:len(m)//2]**

**b=m[len(m)//2:]**

**c=int(a)+int(b)**

**d=c\*\*2**

**if d==n:**

**print("tech number")**

**else:**

**print("not tech number")**

**27.\*\*Armstrong number or not:**

**num=int(input("enter the number:"))**

**sum=0**

**temp=num**

**while temp>0:**

**digits=temp%10**

**sum=sum+digits\*\*3**

**temp=temp//10**

**if sum==num:**

**print("armstrong number")**

**else:**

**print("not armstrong number")**

**\*\*happy number or not:**

**n=int(input("enter the number:"))**

**x=n**

**while x>=10:**

**sum=0**

**while x>0:**

**r=x%10**

**sum=sum+r\*\*2**

**x=x//10**

**x=sum**

**if x==1:**

**print("happy number")**

**else:**

**print("not happy number")**

**\*\*\*prime numbers between n numbers:**

**n1=int(input("enter the number:"))**

**n2=int(input("enter the number:"))**

**for i in range (n1,n2+1):**

**if i>1:**

**fact=0**

**for j in range(1,i):**

**if i%j==0:**

**fact=fact+1**

**if fact==1:**

**print(i)**

**\*\*Calender programm:**

**# Define days of the week**

**days\_of\_week = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"]**

**# Define months and their corresponding number of days**

**months = {**

**"January": 31,**

**"February": 28,**

**"March": 31,**

**"April": 30,**

**"May": 31,**

**"June": 30,**

**"July": 31,**

**"August": 31,**

**"September": 30,**

**"October": 31,**

**"November": 30,**

**"December": 31**

**}**

**# Get user input for the year**

**year = int(input("Enter year: "))**

**# Check if the year is a leap year**

**if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):**

**months["February"] = 29**

**# Define a variable to keep track of the starting day of the week**

**start\_day = 0**

**# Loop through each month**

**for month, num\_days in months.items():**

**print(month, year)**

**print("--------------------")**

**# Print the header with days of the week**

**for day in days\_of\_week:**

**print(day[:2], end=" ")**

**print()**

**# Print the days of the month**

**for i in range(1, num\_days + 1):**

**# Print leading spaces for the first week**

**if i == 1:**

**print(" " \* start\_day, end="")**

**print("{:2}".format(i), end=" ")**

**# Move to the next line if it's the end of the week**

**if (start\_day + i) % 7 == 0:**

**print()**

**# Calculate the starting day of the next month**

**start\_day = (start\_day + num\_days) % 7**

**print("\n")**

\*\*A program to print the number of vowels and number of consonants in the given statement and which is maximum

**stmt=input("enter the string:")**

**x='aeiouAEIOU'**

**y='qwrtyplkjhgfdsazxcvbnmQWRTYPLKJGFDSAZXCVBNM'**

**vowels=0**

**consonants=0**

**for char in stmt:**

**if char in x:**

**vowels=vowels+1**

**elif char in y:**

**consonants=consonants+1**

**else:**

**pass**

**print("number of vowels:",vowels)**

**print("number of cosonants:",consonants)**

**if vowels>consonants:**

**print("vowels are maximum")**

**elif vowels<consonants:**

**print("consonants are maximum")**

**else:**

**print("vowels and consonants are same")**

\*\*Program to find whether two strings have same character in same index and returns the number of matches:

**string1=input("enter the string:")**

**string2=input("enter the string:")**

**count=0**

**for char1,char2 in zip(string1,string2):**

**if char1==char2:**

**count=count+1**

**print("number of matches:",count)**

\*\*Program to print number of words in a line and number of lines in a para

**paragraph ="""This is a sample paragraph.**

**It consists of multiple lines.**

**Each line contains several words.**

**The program will count the number of words in each line.**

**It will also count the total number of lines in the paragraph."""**

**lines=paragraph.split("\n")**

**num\_lines=len(lines)**

**total\_words=0**

**for line in lines:**

**words=line.split()**

**num\_words=len(words)**

**total\_words=total\_words+num\_words**

**print("number of words in line:",num\_words)**

**print("numbers of lines:",num\_lines)**

**print("number of words:",total\_words)**

\*\*Program to find number of sentences starts with "B"

**text='''The apple doesn't fall. ...**

**All that glitters are not gold. ...**

**A picture is worth a thousand words. ...**

**Beggars can't be choosers. ...**

**A bird in the hand. ...**

**Better safe than sorry. ...**

**An apple a day keeps doctor away. ...**

**Blood is thicker than water. ...'''**

**sentences=text.split("...")**

**count\_b\_sen=0**

**for sentence in sentences:**

**if sentence.strip().startswith('B'):**

**count\_b\_sen=count\_b\_sen+1**

**print("number of sentences starts with B :",count\_b\_sen)**

\*\*Write a program that finds whether a given character is present in a string or not. In case it is present it prints the index at which it is present. Do not use built-in find functions to search the character.

**string=input("enter the string:")**

**char=input("enter the character to find in string:")**

**index=-1**

**for i in range(len(string)):**

**if string[i]==char:**

**index=i**

**break**

**if index!=-1:**

**print(f" character is present at index:{index}")**

**else:**

**print(f" character is not present in string")**

**word = input("Enter a word: ")**

**# Normal order**

**sorted\_word = ''.join(sorted(word))**

**print("Word in normal alphabetical order:", sorted\_word)**

**# Reverse order**

**reverse\_sorted\_word = ''.join(sorted(word, reverse=True))**

**print("Word in reverse alphabetical order:", reverse\_sorted\_word)**

\*\*Write a program that accepts a string from user and re displays the same string after removing vowels from it.

**string=input("enter the string:")**

**vowels='aeiouAEIOU'**

**new\_string=""**

**for char in string:**

**if char in vowels:**

**pass**

**else:**

**new\_string=new\_string+char**

**print(new\_string)**

**\*\*COMPOSITE NUMBERS IN LIST:**

**list=[16, 18, 27, 16, 23, 21, 19]**

**result=[]**

**for num in list:**

**count=0**

**for i in range(1,num):**

**if num%i==0:**

**count=count+1**

**if count>2:**

**result.append(num)**

**break**

**print("composite numbers:",result)**

**print("number of composite numbers:",len(result))**

**\*\*print non-composite numbers:**

**list=[26, 28, 47, 26, 43, 51, 29]**

**result=[]**

**for num in list:**

**fact=0**

**for i in range(2,num):**

**if num%i==0:**

**fact=fact+1**

**if fact==0:**

**result.append(num)**

**print("non composite numbers:",result)**

**print("number of non composite:",len(result))**

**\*\*** Find the Mth maximum number and Nth minimum number in an array and then find the sum of it, difference of it and product of it

**arr = [5, 8, 2, 10, 3]**

**arr.sort()**

**m=int(input("enter the mth maximum number:"))**

**n=int(input("enter the nth minimum number:"))**

**max\_num=arr[len(arr)-m]**

**mini\_num=arr[n-1]**

**print(max\_num)**

**print(mini\_num)**

**sum=max\_num+mini\_num**

**difference=max\_num-mini\_num**

**product=max\_num\*mini\_num**

**print("sum:",sum)**

**print("difference:",difference)**

**print("product:",product)**

**\*\***A peak element is an element that is strictly greater than its neighbours. Given a **0-**

**indexed** integer array nums, find a peak element, and return its index. If the array contains

multiple peaks, return the index to **any of the peaks**

**list=[1,2,3,1]**

**for i in range(len(list)):**

**if i==0 and list[i]>list[i+1]:**

**print(i)**

**break**

**elif i==len(list)-1 and list[i]>list[i-1]:**

**print(i)**

**break**

**elif list[i]>list[i-1] and list[i]>list[i+1]:**

**print(i)**

**break**