**ASSIGNMENT – 1**

**Submitted to: Anit James**

**Submitted by: Maria Jacob**

**Roll No: 36**

**INT MCA, SEM 9**

**Subject: Python programming**

**SET-1**

1. **Program to find the GCD of two number**

**CODE:**

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

b=int(input("Enter the second number"))

for i in range(1, min(a,b)+1):

    if a%i==0 and b%i==0:

        gcd=i

print("GCD of", a , "and", b , "is" ,gcd)

**OUTPUT**

Enter a number 8

Enter the second number 10

GCD of 8 and 10 is 2

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Program to find the factorial of a number**

**CODE:**

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

fact=1

for i in range(1,a+1):

    fact=fact\*i

print("Factotrial of" , a , "is : " , fact)

**OUTPUT**

Enter a number 4

Factotrial of 4 is : 24

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Program to generate fibonacci series of n terms**

**CODE:**

a=int(input("Enter a number of terms"))

f1,f2=0,1

f3=f1+f2

print("Fibonacci series of first" , a , "terms is : ")

print(f1)

print(f2)

for i in range(3,a+1):

    print(f3)

    f1=f2

    f2=f3

    f3=f1+f2

**OUTPUT**

Enter a number of terms 6

Fibonacci series of first 6 terms is :

0

1

1

2

3

5

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Program to count the number of vowels.**

**CODE:**

a=input("Enter a string")

count=0

for i in a:

    if i in 'aeiouAEIOU':

        count=count+1

print("Number of vowels : ", count)

**OUTPUT**

Enter a string Python Programming

Number of vowels : 4

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Program to find the sum of all items in a list.**

**CODE:**

list=[1,2,3,4]

sum=0

for i in range(1,5):

    sum=sum+i

print(sum)

**OUTPUT**

10

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Program to print the pyramid.**

**CODE:**

a=int(input("Enter the row of pyramid"))

for i in range(1,a+1):

    for j in range(1,i+1):

        print(i,end='')

    print()

**OUTPUT**

Enter the row of pyramid 5

1

22

333

4444

55555

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Primitive Pythagorean Triads**

**CODE:**

for i in range(1,50):

    for j in range(1,i):

        for k in range(1,j):

            if k\*k+j\*j==i\*i:

                flag=0

                for l in range(2,i):

                    if i%l==0 and j%l==0 and k%l==0:

                        flag=1

                        break

                if flag:

                    continue

                print("a=",k,"b=",j,"c=",i)

**OUTPUT**

a= 3 b= 4 c= 5

a= 5 b= 12 c= 13

a= 8 b= 15 c= 17

a= 7 b= 24 c= 25

a= 20 b= 21 c= 29

a= 12 b= 35 c= 37

a= 9 b= 40 c= 41

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**12. Program to print the Chessboard**

**CODE:**

r=int(input("Enter Row Number : "))

c=int(input("Enter Column Number : "))

while(c in range(1,9) and r in range(1,9)):

    print("Possible Movements in (row,col)")

    if r==1 and c==1:

        print(r,c+1)

        print(r+1,c)

        break

    elif r==1 and c==8:

        print(r,c-1)

        print(r+1,c)

        break

    elif c==1 and r==8:

        print(r-1,c)

        print(r,c+1)

        break

    elif c==8 and r==8:

        print(r-1,c)

        print(r,c-1)

        break

    elif c==1 and r<8:

        print(r,c+1)

        print(r+1,c)

        print(r-1,c)

        break

    elif r==1:

        print(r+1,c)

        print(r,c+1)

        print(r,c-1)

        break

    elif c==8:

        print(r-1,c)

        print(r+1,c)

        print(r,c-1)

        break

    elif r==8:

        print(r-1,c)

        print(r,c+1)

        print(r,c-1)

        break

    else:

        print(r,c-1)

        print(r,c+1)

        print(r+1,c)

        print(r-1,c)

        break

else:

    print("Invalid range for row or column")

**OUTPUT:**

Enter Row Number : 3

Enter Column Number : 4

Possible Movements in (row,col)

3 3

3 5

4 4

2 4

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**14. Count a number in a given list**

**CODE:**

lis=input("Enter a list: ")

list1=list(map(int,lis.split()))

n=int(input("Enter the number to searched: "))

print(lis)

print("The number of occurance of ",n," is ",list1.count(n)," times")

**OUTPUT**

Enter a list: 2 4 6 8 2 4 6 5 3 1 7 9

Enter the number to searched: 2

2 4 6 8 2 4 6 5 3 1 7 9

The number of occurance of 2 is 2 times

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**15. Write a program to get the n (non-negative integer) copies of the first two characters of a given string.**

**CODE:**

str=input("enter a string")

n=int(input("number of copies of first two characters"))

flen=2

if flen>len(str):

    flen=len(str)

substr=str[:flen]

result=""

for i in range (n):

        result=result+substr

print("copy of substring",result)

**OUTPUT**

enter a string maria

number of copies of first two characters2

copy of substring m m

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**16. Check whether a specified value is contained in a group of values**

**CODE:**

str=input("enter a list(value space seperated)")

n= int(input("enter the number to search  "))

list=list(map(int,str.split()))

print(list)

for i in list:

    if n==i:

        print("the  number ",n,"found in list")

        break

else:

    print("the  number ",n,"not found in list")

**OUTPUT**

enter a list(value space seperated)2 4 5 7 6 5

enter the number to search 7

[2, 4, 5, 7, 6, 5]

the number 7 found in list

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**18. Write a program to print all even numbers from given list**

**CODE:**

lis=input("Enter a list(elements space seperated): ")

lis1=list(map(int, lis.split()))

print(lis1)

print("Even numbers upto 237")

for x in lis1:

    if x==237:

        break

    elif x%2==0:

        print(x, end=' ')

**OUTPUT**

Enter a list(elements space seperated): 2 4 6 8 1 3 55 88 9 22 44

[2, 4, 6, 8, 1, 3, 55, 88, 9, 22, 44]

Even numbers upto 237

2 4 6 8 88 22 44

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**19. Write a python program to get the lcm of two positive integers**

**CODE:**

def lcm(x, y):

gcd = 1

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

if x % i == 0 and y % i == 0:

gcd = i

lcm = (x \* y) // gcd

return lcm

x = int(input("Enter the first number: "))

y = int(input("Enter the second number: "))

print("The LCM of {} and {} is {}".format(x, y, lcm(x, y)))

**OUTPUT**

Enter the first number: 10

Enter the second number: 5

The LCM of 10 and 5 is 10

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**21. Write a python program to get a string made of the first 2 and the last 2 characters from a given string.**

**CODE:**

def get\_first\_and\_last\_two\_chars(string):

  if len(string) < 2:

    return ""

  first\_two\_chars = string[:2]

  last\_two\_chars = string[-2:]

  return first\_two\_chars + last\_two\_chars

string = input("Enter a string: ")

first\_and\_last\_two\_chars = get\_first\_and\_last\_two\_chars(string)

print(f"The string made of the first 2 and the last 2 chars from '{string}' is '{first\_and\_last\_two\_chars}'.")

**OUTPUT**

Enter a string: Python programming

The string made of the first 2 and the last 2 chars from 'Python programming' is 'Pyng'.

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**22. Write a python program to add ‘ing’ at the end of a given string.**

**CODE:**

str1=input("Enter a string :")

length= len(str1)

if length>2:

    if str1[-3:]=='ing':

        str1+='ly'

    else:

        str1+= 'ing'

print("New String : ", str1)

**OUTPUT**

Enter a string :Python Programming

New String : Python Programmingly

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**24. Write a python function that takes a list of word and returns the length of the longest one.**

**CODE:**

lis=input("Enter a list with some strings :")

words\_list=lis.split()

word\_len=[]

for n in words\_list:

    word\_len.append((len(n),n))

    print(word\_len)

    word\_len.sort()

    print(word\_len)

print("Longest Word : ",word\_len[-1][1])

**OUTPUT**

Enter a list with some strings :maria python apple orange

[(5, 'maria')]

[(5, 'maria')]

[(5, 'maria'), (6, 'python')]

[(5, 'maria'), (6, 'python')]

[(5, 'maria'), (6, 'python'), (5, 'apple')]

[(5, 'apple'), (5, 'maria'), (6, 'python')]

[(5, 'apple'), (5, 'maria'), (6, 'python'), (6, 'orange')]

[(5, 'apple'), (5, 'maria'), (6, 'orange'), (6, 'python')]

Longest Word : python

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**25. Write a python program to remove the characters which have odd index values of a given string.**

**CODE:**

input\_string = input("Enter a string: ")

result\_string = ""

for index, char in enumerate(input\_string):

    if index % 2 == 0:

        result\_string += char

print("String with odd-index characters removed:", result\_string)

**OUTPUT**

Enter a string: Python programming

String with odd-index characters removed: Pto rgamn

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**27. Write a python program to remove the characters which have odd index values of a given string.**

**CODE:**

input\_sequence = input("Enter comma seperated strings")

words = input\_sequence.split(',')

cleaned\_words = [word.strip().lower() for word in words]

unique\_sorted\_words = sorted(set(cleaned\_words))

print("Unique strings: ")

print(', '.join(unique\_sorted\_words))

**OUTPUT**

Enter comma seperated strings blue,black,orange,blue,red,yellow,red,green,blue

Unique strings:

black, blue, green, orange, red, yellow

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**28. Python program to Count a number of strings where the string length is 2 or more and the first and last character are same from a given list of string**

**CODE:**

lis=input("Enter a list :")

words=list(lis.split())

ctr=0

for word in words:

    if len(word)>1 and word[0] ==word[-1]:

        ctr+=1

print("Count=",ctr)

**OUTPUT**

Enter a list : cat bob pop dog am pat abba

Count= 3

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**30. Write a python program to check a list is empty or not.**

**CODE:**

l=input("Enter a list: ")

lis=list(l.split())

if not lis:

    print("Empty List")

else:

    print("List isnot empty")

    print(lis)

print("The list is not empty.")

**OUTPUT**

Enter a list: 1 2 3 4 5 6

List isnot empty

['1', '2', '3', '4', '5', '6']

The list is not empty.

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**31. Write a python program to find the list of words that are longer than n from a given list of words.**

**CODE:**

s=input("Enter list of seperated: ")

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

t=s.split()

word\_length=[]

for x in t:

    if len(x)>n:

        word\_length.append(x)

print("Words with length greater than",n,"=",word\_length)

**OUTPUT**

Enter list of seperated: apple mango orange grapes bat rat

Enter length: 3

Words with length greater than 3 = ['apple', 'mango', 'orange', 'grapes']

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**33. Write a python program to generate a 3\* 4\* 6 D array whose each element is \*.**

**CODE:**

a = [[ ['\*' for col in range(6)] for col in range(4) ] for row in range(3) ]

print("3\* 4\* 6 D array is : ",a)

**OUTPUT**

3\* 4\* 6 D array is :

[[['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*']], [['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*']], [['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*'], ['\*', '\*', '\*', '\*', '\*', '\*']]]

**35. Write a python program to generate and print a list of first and last 5 elements where the values are the square of numbers between 1 and 30.**

**CODE:**

l = list()

for i in range(1,21):

    l.append(i\*\*2)

print(l[:5])

print(l[-5:])

**OUTPUT**

[1, 4, 9, 16, 25]

[256, 289, 324, 361, 400]

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**38. Write a python program to generate and print a dictionary that contains number between 1 and n in the form (x, x\*x)**

**CODE:**

n=int(input("enter limit: "))

d=dict()

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

    d[i]=i\*i

print(d)

**OUTPUT**

enter limit: 4

{1: 1, 2: 4, 3: 9, 4: 16}

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**41. Write a python program to convert temperature to and from Celsius, foresheet.**

**CODE:**

temp = input("Enter the temperature to convert : ")

degree = int(temp[:-1])

i\_convention = temp[-1]

if i\_convention.upper() == "C":

  result = int(round((9 \* degree) / 5 + 32))

  o\_convention = "Fahrenheit"

elif i\_convention.upper() == "F":

  result = int(round((degree - 32) \* 5 / 9))

  o\_convention = "Celsius"

else:

  print("Enter proper convention!")

  quit()

print ("The temperature in", o\_convention, "is", result, "degrees")

**OUTPUT**

Enter the temperature to convert : 25C

The temperature in Fahrenheit is 77 degrees

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**43. Write a python program that accepts a word from the user and reverse it.**

**CODE:**

word= input("enter a word to reverse: ")

for char in range(len(word)- 1, -1,-1):

    print(word[char], end="")

**OUTPUT**

enter a word to reverse: maria

airam

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**44. Write a python program that counts odd and even numbers from a list.**

**CODE:**

lis=input("Enter positive integers with space seperation: ")

numbers=list(map(int,lis.split()))

odd\_count=0

even\_count=0

for i in numbers:

    if not i%2:

        even\_count+=1

    else:

        odd\_count+=1

print("Number of even numbers: ", even\_count)

print("Number of odd numbers: ", odd\_count)

**OUTPUT**

Enter positive integers with space seperation: 23 4 55 1 2 65 45 34

Number of even numbers: 3

Number of odd numbers: 5

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**47. Write a python program that accepts 3 sequence of comma separated 4 digits binary numbers as its input and print the numbers that are divisible by 5 in a comma separated sequence.**

**CODE:**

items=[]

n=input("Enter binary numbers with comma seperated : ")

n1=list(n.split(','))

for p in n:

    x=int(p,2)

    if not x%5:

        items.append(p)

print(','.join(items))

**OUTPUT**

Enter binary numbers with comma seperated : 0101,1100,1111,1010

0101,1111,1010

**49.Write a python program to find numbers between 100 and 400 where each digit of a number. The numbers obtained should be printed in a comma-separated sequence.**

**CODE:**

items=[]

for i in range(100, 401):

    s=str(i)

    if (int(s [0])%2==0) and (int(s [1])%2==0) and int(s [2])%2==0:

        items.append(s)

print(",".join(items))

**OUTPUT**

200,202,204,206,208,220,222,224,226,228,240,242,244,246,248,260,262,264,266,268,280,282,284,286,288,400

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**SET-2**

1. **Write a python function to check whether a number if even or odd.**

**CODE:**

def is\_even\_or\_odd(number):

    if number % 2 == 0:

        print("The number ",number, "is even")

    else:

        print("The number ",number, "is odd")

result = int(input("Enter the number : "))

is\_even\_or\_odd(result)

**OUTPUT**

Enter the number : 6

The number 6 is even

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python program to calculate the sum of three given numbers, if the values are equal then return thrice of their sum.**

**CODE:**

def calculate\_sum\_or\_triple(num1, num2, num3):

    if num1 == num2 == num3:

        return 3 \* (num1 + num2 + num3)

    else:

        return num1 + num2 + num3

num1 = float(input("Enter the first number: "))

num2 = float(input("Enter the second number: "))

num3 = float(input("Enter the third number: "))

result = calculate\_sum\_or\_triple(num1, num2, num3)

print("Result:", result)

**OUTPUT**

Enter the first number: 3

Enter the second number: 3

Enter the third number: 3

Result: 27.0

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **write a python function to get a new string from a given string where "Is" has been added to the front. i the given string already begins with "is" then return the string unchanged**.

**CODE:**

def add\_is\_to\_string(input\_string):

    if input\_string.lower().startswith("is"):

        return input\_string

    else:

        return "Is" + input\_string

user\_input = input("Enter a string: ")

new\_string = add\_is\_to\_string(user\_input)

print("Modified string:", new\_string)

**OUTPUT**

Enter a string: Good Morning

Modified string: IsGood Morning

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python program to get a string which is n copies of a given string.**

**CODE:**

def generate\_copies(input\_string, n):

    return input\_string \* n

user\_string = input("Enter a string: ")

num\_copies = int(input("Enter the number of copies: "))

result\_string = generate\_copies(user\_string, num\_copies)

print("Result:" , result\_string)

**OUTPUT**

Enter a string: Python Program

Enter the number of copies: 3

Result: Python Program Python Program Python Program

1. **Write a python function that will return true if the two given integer values are equal or their sum or difference is 5.**

**CODE:**

def test\_num5(x,y):

    eq,dif5,sum5,no=False,False,False,False

    if x==5 or y==5:

        eq=True

    elif abs(x-y)==5:

        dif5=True

    elif (x+y)==5:

        sum5=True

    else:

        no=True

    return eq,dif5,sum5,no

a=int(input("Enter first integer: "))

b=int(input("Enter second integer: "))

e,d,s,n=test\_num5(a,b)

if e:print(a,"and" ,b, "are equal : ",e)

if d:print("Difference of" , a , "and", b ,"is", 5 ,"is", d)

if s:print("Sum of" ,a, "and", b ,"is", 5 ,"is", s)

if n:print("Sum of numbers is not 5\n Difference of numbers is not 5\n numbers are not equal")

**OUTPUT**

Enter first integer: 6

Enter second integer: 8

Sum of numbers is not 5

Difference of numbers is not 5

numbers are not equal

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python function to generate all the factors of a number**

**CODE:**

def find\_factors(number):

factors = []

for i in range(1, number + 1):

if number % i == 0:

factors.append(i)

return factors

user\_input = input("Enter a positive integer: ")

user\_input = int(user\_input)

if user\_input <= 0:

print("Please enter a positive integer.")

else:

result = find\_factors(user\_input)

print("Factors of", user\_input, "are:", result)

**OUTPUT**

Enter a positive integer: 12

Factors of 12 are: [1, 2, 3, 4, 6, 12]

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python function to find the sum of digit of a number.**

**CODE:**

def sum\_of\_digits(number):

    total = 0

    while number > 0:

        total += number % 10

        number //= 10

    return total

user\_input = input("Enter a number: ")

user\_input = int(user\_input

result = sum\_of\_digits(user\_input)

print("Sum of digits:", result)

**OUTPUT**

Enter a number: 1234

Sum of digits: 10

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python function to concatenate two strings.**

**CODE:**

def concatenate\_strings(str1, str2):

return str1 + str2

user\_input1 = input("Enter the first string: ")

user\_input2 = input("Enter the second string: ")

result = concatenate\_strings(user\_input1, user\_input2)

print("Concatenated string:", result)

**OUTPUT**

Enter the first string: maria

Enter the second string: jacob

Concatenated string: mariajacob

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

1. **Write a python function called compare which takes two strings s1 and s2 and an integer n as arguments. The function should return True if first n character of both the strings are same else the function should return false.**

**CODE:**

def compare(s1, s2, n):

    if len(s1) < n or len(s2) < n:

        return False

    for i in range(n):

        if s1[i] != s2[i]:

            return print("Substrings not are equal ")

return print("Substrings are equal ")

user\_input1 = input("Enter the first string: ")

user\_input2 = input("Enter the second string: ")

user\_input\_n = int(input("Enter the value of n: "))

result = compare(user\_input1, user\_input2, user\_input\_n)

print(result)

**OUTPUT**

Enter the first string: Programming

Enter the second string: program

Enter the value of n: 4

Substrings not are equal

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>

**14. Write a python program to display Fibonacci series using recursion.**

**CODE:**

def fibonacci(n):

    if n <= 0:

        return []

    elif n == 1:

        return [0]

    elif n == 2:

        return [0, 1]

    else:

        fib\_sequence = fibonacci(n - 1)

        fib\_sequence.append(fib\_sequence[-1] + fib\_sequence[-2])

        return fib\_sequence

user\_input = int(input("Enter the number of Fibonacci terms to display: "))

result = fibonacci(user\_input)

print("Fibonacci Sequence:", result)

**OUTPUT**

Enter the number of Fibonacci terms to display: 5

Fibonacci Sequence: [0, 1, 1, 2, 3]

PS C:\Users\HP\Documents\Maria\_AES\SEM\_9\python>