LIST OF LAB QUESTION(S)

PROGRAMMING IN PYTHON : ICT 160

[APPLICABLE FOR BATCH CODES: RA B1-B ; AIDS B2-A,B, RA B2-A,B ; AIML B2-A; IIOT B1-A,B]

Q1) Write a Program to perform string manipulation operations using set of pre-defined functions such as :

a) Find()
b) Upper()
c) Len()
d) Max() and Min()
e) Fetching a specific content from the String

Q2) Write a Program to perform to test and check the mathematical functions such as :

a) Ceil()
b) Sqrt()
c) Pow()
d) Factorial()

Q3) Write a Program that receive a number as input from user and returns if it odd or even number.

Q4) Write a Program that receive input from the user to calculate the Area of Triangle

Q5) Write a Program that receive input from the user to calculate the Area of Square

Q6) Write a Program that receive input from the user to calculate the Area of Rectangle

Q7) Write a Program to check if the input string is Palindrome or not

Q8) Write a Program that receives marks of a students for a subject as input and assign the grades A||B||C||D||E||F

Q9) Write a Program to compute the GCD of the two numbers.

Q10) Write a Program to check if the given number is Armstrong number or not. Example of Armstrong number are :- 153, 370, 371 etc.

Q11) Write a Program to check if the input year is a leap year or not

Q12) Write a Program to computer factorial of a given number

Q13) Write a Program to generate Fibonacci series till 100.

Q14) Write a Program to create a two list and perform the following operation's :

1) Add the Elements of the two list.

2) Compare the contents of the two list.

3) to find the number of the elements in the list.

4) Sort the elements of the list

5) Reverse the contents of the List.

Q15) Write a Program to create and display the content of the tuple. Initialize the tuple with the name of the cities. Display content of the tuple along with name/index positions of the cities.

Q16) Write a program to create an Array of Even numbers till 14. Display the contents of array, compute the length of array and also show how to delete a element from the desired position from the array.

Q17) Using Filter function, write a program to filter the elements which are greater than 9.

Q18) Using Filter function, write a program to display multiple of 5 from a given array.

Q19) Write a Program to create a file called “Input.txt”, perform the write/read operation in it with a string “Computer Science”.

Q20) Write a Program to create a file called “Input.txt”, initialize it with a string of your choice and perform the read operation to read only the first 3 characters from the file.

Q21) Using NumPy, write a program to create 1 Dim Array, load it with numbers, and perform the operation of Iteration and Slicing on it.

Q22) Using NumPy, write a program to create Multi-Dim Array, load it with the numbers and display the content of it.

Q23) Using NumPy, write a program to create two 1 Dim Array and perform the operation of Iteration, Sorting the contents of array and concatenating the contents of the array.

Q24) Using NumPy, initialize the array and display their dimensionality.

Q25) Using Panda, create a DataFrame, initialize it with the contents such as : your Enrollment Number and Name and display them.

Q26) Create 2 array, using the MatPlotLib, plot the graph with the content of the two array, with coordinates plotting on X axis and Y axis.

Q27) Create a .csv file(with contents like : Age, Weight and BMI). Read the content of the file and using Panda and MatPlotLib, plot the graph.

Q28) Create a .csv file(with contents like : Age, Weight and BMI). Read the content of the file and using Panda and MatPlotLib, plot the histogram

Q29) Write a Program to create a class called ‘Student’ with fields such as : Enrollment Number, USS Name, Branch Name, Student Name etc. Instantiate a class and make a call to user defined function to display the details of students.

Q30) Define Employee Class with fields such as Employee ID and Employee Name. Instantiate the class, invoke the constructor and make a call to user defined function to display the information about employee.

PROGRAM-1

- QUESTION-1 : Write a program to perform string manipulation operation using set of pre-defined functions such as :

(i) find()
(ii) upper()
(iii) Len()
(iv) Max()
(v) Min()
(vi) fetching specific content from string

- CODE :

```
1. String = 'ggsipueastcampus'
2.
3 #1.find()
4. print (string.find('i'))
5.
6. #2. upper()
7. print (string.upper())
8.
9. # 3. lower()
10. print (string.lower())
11.
12 # 4. len()
13 print (string.len())
14
```

```
#5. max()
print(string.max())

#6. min()
print(string.min())
```

PROGRAM-2

- QUESTION-2 : Write a Program to perform to test and check the mathematical functions such as:

(i) ceil()
(ii) sqrt()
(iii) Pow()
(iv) factorial().

- CODE :

(i) ceil() :

```
import math.

n = float(input('Enter a number:'))

print(math.ceil(n))
```

(ii) sqrt()

```
import math.

num = float(input('Enter a number:'))
print(math.sqrt(num))
```

(iii) Pow ()

```
1. import math.
2.
3. num = float (input ('Enter a number :'))
4.  a = float (input ('Enter a number :'))
5.
6. print (pow(num, a))
```

(iv) factorial ().

```
1. import math.
2.
3. num = float (input ('Enter a number))
4. print (math. factorial (num))
```

DELTA Pg No.
Date / /

PROGRAM-3

- QUESTION 3 : W.A.P that receive a number as input from user and returns if it odd or even number

- CODE :

```
1. num = int(input('Enter any +ve number:'))
2.
3. if (num % 2) == 0:
4.
5.     print('The number is even')
6.
7. else :
8.
9.     print('The provided number is odd').
```

PROGRAM-4

- QUESTION:4 - W.A.P that receive input from the user to calculate area of square.

- CODE:

```
1. n = int(input("Enter side of square:"))
2.
3. area = n*n
4.
5. print("\n Area of square =", area).
```

DELTA Pg No.
Date / /

PROGRAM-5

- QUESTION-5 : W.A.P that receive input from the user to calculate the Area of triangle.

- CODE :

```
1. x = float(input('Enter first side :'))
2. y = float(input('Enter second side :'))
3. z = float(input('Enter third side :'))
4.
5. s = (x+y+z)/2.
6.
7. area = (s*(s-x)*(s-y)*(s-z))**0.5
8.
9. print('The area of the triangle is : %.0.2f' %area)
```

PROGRAM-6

- QUESTION-6 : W.A.P that receive input from the user to calculate Area of Rectangle

- CODE :

```
1. l = float(input('Enter length :'))
2. b = float(input('Enter breadth :'))
3.
4. Area = l*b
5.
6. print("Area of Rectangle is :%.2f", %Area)
```

DELTA Pg No.
Date / /

PROGRAM-7

- QUESTION-7: Write a program to check if the input string is Palindrome or not.

- CODE:

```
1.  n = input("Enter the string")
2.  flag = 0
3.
4.  i = 0
5.  j = len(n) - 1
6.  while i > j:
7.      if n[i] == n[j]:
8.          pass
9.      else:
10.         flag = 1
11.         break
12.     i = i + 1
13.     j = j - 1
14. if flag == 1:
15.     print(n, "is not a Palindrome")
16.
17. else:
18.     print(n, "is a palindrome)
```

PROGRAM-8

- QUESTION-8: W.A.P that receive marks of a student for a subject as input & assign the grades A||B||C||D||E.||F

- CODE:

```
1. S1 = int(input("Enter marks: "))
2. S2 = int(input("Enter marks:"))
3. S3 = int(input("Enter marks:"))
4. S4 = int(input("Enter marks:"))
5. S5 = int(input("Enter marks:"))
6.
7. avg = (S1+S2+S3+S4+S5)/5
8.
9. if(avg>=90):
10.     print("Grade: A")
11.
12. if(avg>=80 and avg<90):
13.     print("Grade: B")
14.
15. if(avg>=70 and avg<80):
16.     print("Grade: C")
17.
18. if(avg>=60 and avg<70):
19.     print("Grade: D")
```

```
20 else :
21         print("Grade : F")
```

DELTA Pg No.
Date / /

PRACTICAL - 9

- QUESTION-9 : Write a program to compute the GCD of the two numbers.

- CODE :

```
num1 = int (input ('Enter the first number:'))
num2 = int (input ('Enter the second number:'))

a = num1
b = num2

while (num2 !=0):

    temp = num2
    num2 = num1 % num2
    num1 = temp
gcd = num1

print ('gcd of two number is:', gcd)
```

PROGRAM-10

- QUESTION-10 : Write a Program to check if the given number is Armstrong number or not.

- CODE :

```
1. num = int (input (" Enter a no :"))
2.
3  sum = 0
4.
5  temp = num
6  while temp > 0
7.          digit = temp % 10
8.          sum + = digit ** 3
9           temp // = 10
10.
11. if num == sum
12.     print (num, "is an Armstrong number")
13.
14. else:
15.     print (num," is not an Armstrong number")
```

PROGRAM-11

- QUESTION-11 : Write a program to check if input year is a leap year or not.

- CODE :

```
1. year = int(input("Enter year :"))
2.
3. if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
4.
5.     print(year, "is a leap year")
6.
7. else:
8.
9.     print(year, "is not a leap year")
```

PROGRAM-11

- QUESTION-11 : Write a program to check if input year is a leap year or not.

- CODE :

```
1. year = int(input("Enter year :"))
2.
3. if (year %. 4 == 0 and year %. 100 != 0) or (year %. 400 == 0)
4.
5.     print(year, "is a leap year")
6.
7. else :
8.
9.     print(year, "is not a leap year")
```

PROGRAM-12

- QUESTION-12: Write a program to compute factorial of a given number

- CODE:

```
1. number = 5
2.
3. factorial = 1.
4.
5. if number < 0:
6.     print(" factorial doesn't exist ")
7.
8. elif number == 0:
9.     print (" The factorial of 0 is 1")
10.
11. else:
12.     for i in range(1, num+1)
13.
14.         factorial = factorial * i
15.
16.     print (" The factorial of", num, "is", factorial)
```

PROGRAM-13

- QUESTION-13: Write a program to generate fibonacci series.

- CODE:

```
1.  n = int(input("Enter the value:"))
2.
3.  a=0
4.  b=0
5.  sum=0
6.  count = 1.
7.
8.  Print("fibonacci series", end=" ")
9.
10. while (count <=n):
11.
12.         Print(sum, end=" ")
13.         count +=1
14.         a = b
15          b= sum
16          sum = a+b.
```

PROGRAM-14

- QUESTION-14: W.A.P to create a two list and perform the following actions:

(i) Add the elements of two list
(ii) compare the elements of two list
(iii) to find the number of elements in list.
(iv) sort the number of elements in list
(v) Reverse list in python.

- CODE:

```
# initializing list

list1 = [5, 15, 50, 18]
list2 = [15, 5, 18, 50]

#1. adding two list
list3 = list1 + list2
print (list3)

#2. compare two list
a = list1
b = list2

```

```
14. if (a==b):
15.     print("list1 and list2 are same")
16.
17. else:
18.     print("list1 and list2 are not same")
19.
20 #3. no. of elements in list
21. elements1 = len(list1)
22. elements2 = len(list2)
23. print("elements in list1 are:", elements1)
24. print("elements in list2 are:", elements2)
25.
26. #4. sorting of list
27. list1.sort()
28. list2.sort()
29. print(list1)
30. print(list2)
31.
32. #5. reversing of list
33. list1.reverse()
34 list2.reverse()
35. print(list1)
36 print(list2)
```

PROGRAM-15

- QUESTION-15: W.A.P to generate and print the data dictionary with your own credentials: Name, Enroll No, Phone number, email id etc

- CODE:

```
1. my_dictionary = {'Name': 'Aman', 'Enrollment No':
2.                  024, 'Phone number': 9912347890,
3.                  'email id': 'xyz@gmail.com'}
4.
5. print(my_dictionary['Name'])
6.
7. print(my_dictionary['Enrollment No'])
8.
9. print(my_dictionary['phone number'])
10.
11. print(my_dictionary['email id'])
```

PROGRAM - 16

- QUESTION - 16 : W.A.P to modify the contents of the pre-defined Data Dictionary by showcasing a change in the phone number and address.

- CODE :

```
my_dictionary = {'Name': 'Aman', 'Enrollment No': 24 , 'Phon
                 number': 9912347890 , 'address': 'New
                 delhi'}

print(my_dictionary['Name'])
print (my_dictionary['Enrollment No'])
print (my_dictionary['Phone number'])
print (my_dictionary['address'],'\n')
print (my_dictionary, '\n')

my_dictionary['Phone number'] = 8800207682
my_dictionary['address'] = 'Agra
Print (my_dictionary)
```

PROGRAM-17

* QUESTION-17: Using filter function, W.A.P to filter the elements which are greater than 9.

- CODE:

```
1. def check (x):
2.
3.     if x>9:
4.         return x.
5.
6. result = filter (check, [1, 2, 5, 11, 12, 16, 45, 8])
7. print (list (result))
```

DELTA Pg No. Date / /

PROGRAM-18

- QUESTION-18 : Using filter function, W.A.P to display multiple of 5 from a given array.

- CODE :

```
1. def mof5(val):
2.
3.     if val%.5 == 0:
4.         return val
5.
6. result = filter (mof5, (1,58, 50,25, 98, 12, 49, 225, 1000))
7. result = list(result)
8.
9. print(result)
```

DELTA Pg No.
Date / /

PROGRAM-19

- QUESTION-19: W.A.P to create a file called "input.txt", perform the write/read operation in it with a string "Computer Science"

- CODE:

```
1. f = open("input.txt", "x")
2. f.write("computer science")
3. f.close
4.
5 f = open("input.txt", "r")
6. print(f.(read())
7 f.close
```

DELTA Pg No.
Date / /

PROGRAM-20

- QUESTION-20 : W.A.P to create a file called "input.txt", initialize it with a string of your choice and perform the read operation to read only the first 3 characters from the file.

- CODE :

```
1. f = open ("input.txt", "w")
2. f.write ("Hello World!")
3. f.close()
4.
5. f = open ("input.txt", "r")
6. print(f.read(3))
7. f.close()
```

PROGRAM-21

- QUESTION-21: Using Numpy, write a program to create 1Dim Array, load it with numbers, & perform the operation of Iteration & slicing on it

- CODE:

```
1. import numpy as np
2.
3. arr = np.array([10,20,30,40])
4. print(arr.dtype)
5.
6. for x in arr:
7.     print(x)
8. print(arr[1])
9. print(arr[1:3])
10. print(arr[2] + arr[3])
```

PROGRAM-22

- QUESTION-22: Using Numpy, O.A.P to create Multi-Dim Array, load it with the numbers and display the content of it.

- CODE:

```
1. import numpy as np
2.
3. arr1 = np.array([[1,2,3,4],[5,6,7,8]])
4.
5. print(arr1)
```

DELTA Pg No.
Date / /

PROGRAM-23

- QUESTION-23 : Using NumPy, W.A.P to create two 1 Dim Array & perform the operation of Iteration, sorting the contents of Array and concatenating the contents of array.

- CODE :

```
1. import numpy as np
2.
3. arr1 = np.array([1,2,3,4])
4.  x = np.where(arr1 == 2)
5. print(np.sort(arr1))
6.
7. # Displaying index position
8. print(x).
9.
10. arr2 = np.array([4,5,6])
11.
12 # Array concatenation
13. arr = np.concatenate((arr1, arr2))
14. print(arr)
```

PROGRAM-24

- QUESTION-24 : Using Numpy, initialize the array and display their dimensionality

- CODE :

```
import numpy as np

a = np.array(50)
b = np.array([10,20,30,40,50])
c = np.array([[10,20,30],[40,50,60]])
d = np.array([[10,20,30],[40,50,60],[[10,20,30];[40,50,60]
5
print(a.ndim)
print(b.ndim)
print(c.ndim)
print(d.ndim)
```

DELTA Pg No. Date / /

PROGRAM-25

- QUESTION 25 : Using Panda, Create a Data frame, initialize it with contents such as: your Enrollment Number and Name & display them

- CODE :

```
1. import panda as pd
2.
3. my data set = {'student name': ["Avi','Varun','Dhruv']
4.                'Enrollment no': [4,3,2]}
5.
6. num = pd.data frame (my data set)
7. print (num).
```

DELTA Pg No.
Date / /

PROGRAM-26

- QUESTION-26: Create 2 array, using the MatPlot lib, plot the graph with the content of the two array with coordinates plotting on x axis & y-axis

- CODE:

```
1. import matplotlib.pyplot as plt
2.
3. import numpy as np
4.
5. xp = np.array([0,2])
6. yp = np.array([0,250])
7.
8. plt.plot(xp,yp)
9.
10. Plt.show()
```

DELTA Pg No. Date / /

PROGRAM-27

- QUESTION-27 : Create a.csv file (with contents like : Age, weight & BMI). Read the content of the file using Panda & Matplob, plot the graph.

- CODE :

```
1. import pandas as pd
2  import matplotlib.pyplot as plt
3
4. df = pd.read_csv('C:\\Users\\Vipul\\Documents\\data1.csv
5.
6. df.plot()
7  plt.show()
```

PROGRAM-28

- QUESTION-28: Create a.csv file (with contents like 'Age, weight, BMI). Read the content of the file using Panda & matplot lib, plot the histogram

- CODE:

```
1 import pandas as pd
2. import matplotlib.pyplot as plt
3.
4. df = pd.read_csv('C:\\Users\\Vipul\\Documents\\data.csv')
5.
6 df.plot(kind='hist')
7. plt.show.
```

DELTA Pg No. Date / /

PROGRAM-29

- QUESTION-29: W.A.P to create a class called 'student' with fields such as: Enroll.no, USS Name, Branch Name, student name etc. Instantiate a class & make a call to user defined function to display.

- CODE:

```
Class student:

    def init (self, name, enroll no , Branch , USSName):
    self.name = name
    self. Enroll no = Enrollno
    self. Branch = Branch
    self. USSName = USSname

Student = student(
    "Vipul",
    "2419011721'
    " USAR"
    " IIOT"
)
print(student.name)
print (student. Enroll no)
print (student. Branch)
print (student. USSName)
```

PROGRAM-30

- QUESTION-30 : Define Employee class with fields such as Employee ID, name, Instantiate the class, invoke the constructor & make a call to user defined function to display the information about employee

- CODE :

```
1. class Employee :
2.
3.     def __init__(self.name, id):
4.         self.name = name
5.         self.id = id.
6.
7.     def display.Employee(self):
8.         print("Name :", self.name, ", id:", self.id)
9.
10. emp1 = Employee("Avi", 2248)
11. emp2 = Employee("Varun, 3275)
12. emp1. display Employee()
13. emp2. display Employee().
```