**DAA Assignments(python programs)**

**Date :- 08/06/2024**

1. **Find a python program to find minimum and maximum in two numbers**

Code-

a = int(input("Enter value of a: "))

b = int(input("Enter value of b: "))

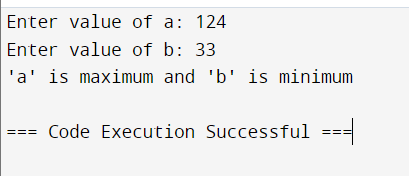
if a > b:

print("'a' is maximum and 'b' is minimum")

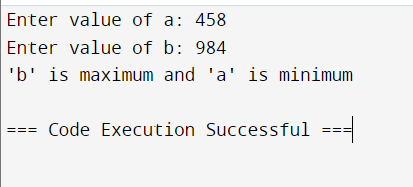
else:

print("'b' is maximum and 'a' is minimum")

output-1



output-2



1. **write a python program to swap two numbers in a list**

**code-**

def swapPositions(lst, pos1, pos2):

lst[pos1], lst[pos2] = lst[pos2], lst[pos1]

return lst

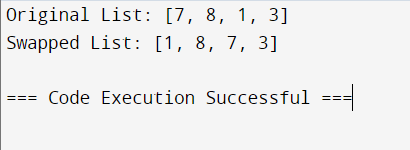
List = [7, 8, 1, 3]

print("Original List:", List)

pos1, pos2 = 1, 3

print("Swapped List:", swapPositions(List, pos1-1, pos2-1))

output-1



1. **Write a python program to find the length of string in four ways**

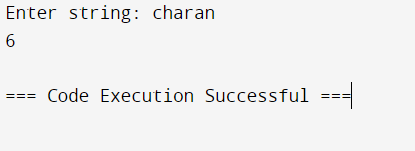
**1st method-**

**Code-**

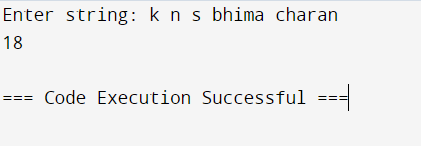
**A= input(“enter string=”)**

**Print(len(a))**

**Output-1**



**Output-2**



**2nd method-**

**Code-**

def findLen(s):

counter = 0

for i in s:

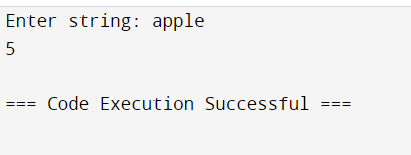
counter += 1

return counter

s = input("Enter string: ")

print(findLen(s))

output-1



1. **Write a python program to reverse a list in python**

**Code-**

lst = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

rev\_lst = []

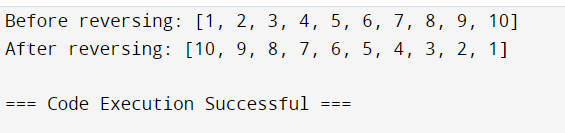
print("Before reversing:", lst)

for i in range(len(lst) - 1, -1, -1):

rev\_lst.append(lst[i])

print("After reversing:", rev\_lst)

output-



**Date :- 14/06/2024**

**5. Write a python program to check whether the given string is a**

**palindrome or not.**

**Code-**

def isPalindrome(s):

return s == s[::-1]

s = input("Enter a string: ")

ans = isPalindrome(s)

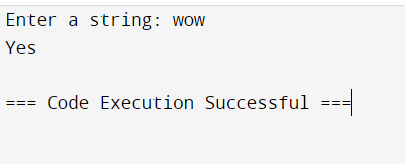
if ans:

print("Yes")

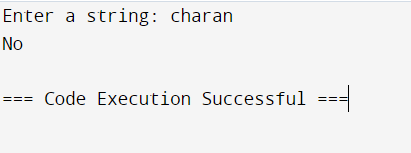
else:

print("No")

output-1



output-2



1. **Write a program to remove ith character from a string**

**Code-**

**def remove\_char(s, i):**

**if 0 <= i < len(s):**

**return s[:i] + s[i+1:]**

**else:**

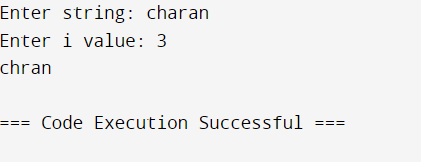
**return s**

**string = input("Enter string: ")**

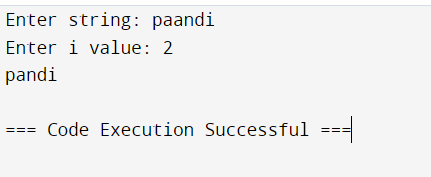
**i = int(input("Enter i value: "))**

**print(remove\_char(string, i-1))**

**output-1**



**output-2**



**Date :- 15/06/2024**

**7. Write a python program to check whether the given number is prime or**

**not.**

Code-

def prime(num):

if num <= 1:

return False

for i in range(2, int(num\*\*0.5) + 1):

if num % i == 0:

return False

return True

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

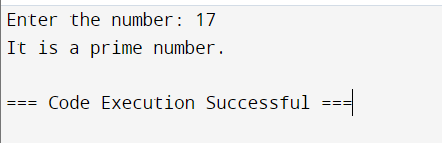
if prime(num):

print("It is a prime number.")

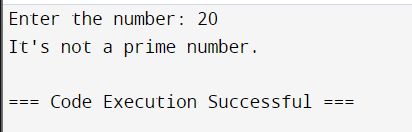
else:

print("It's not a prime number.")

output-1



output-2



**8. write a python program to perform arithmetic operations such as**

**addition,multiplication on complex numbers**

**code-**

def complexoperation(num1, num2):

addition = num1 + num2

subtraction = num1 - num2

multiplication = num1 \* num2

division = num1 / num2

return addition, subtraction, multiplication, division

num1 = complex(input("Enter num 1: "))

num2 = complex(input("Enter num 2: "))

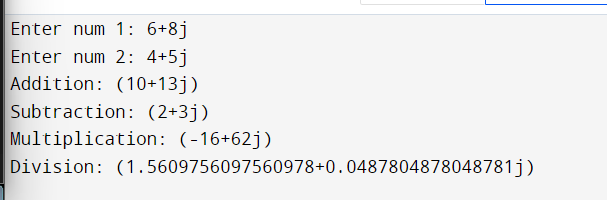
results = complexoperation(num1, num2)

print("Addition:", results[0])

print("Subtraction:", results[1])

print("Multiplication:", results[2])

print("Division:", results[3])

output-1 

date:- 17/06/2024

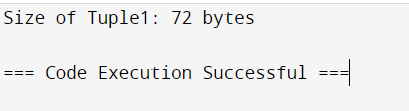
9. program to find size of tuple.

Code-

Tuple1 = ("A", 1, "B", 2, "C", 3)

print("Size of Tuple1: " + str(Tuple1.\_\_sizeof\_\_()) + " bytes")

output-



**10.Find the sum of elements of the tuple**

**Code-**

**def summation(tup):**

**count = 0**

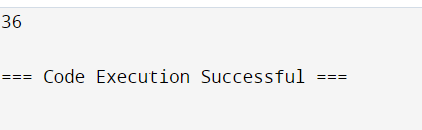
**for i in tup:**

**count += i**

**return count**

**tup = (1, 2, 3, 4, 5, 6,7,8)**

**print(summation(tup))**



**11.create a list of tuples from given list having number and its square in each**

**tuple.**

**Code-**

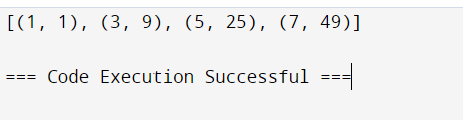
def squareoflist(li):

result = [(num, num\*\*2) for num in li]

return result

li = [1, 3, 5, 7]

print(squareoflist(li))

output- 

12. Write a program to find whether the given number is even or odd.

Source code-

def check\_even\_odd(number):

if number % 2 == 0:

print(f"{number} is even.")

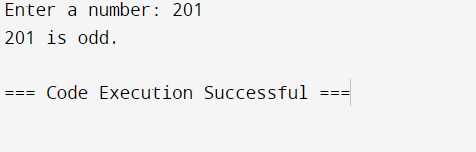
else:

print(f"{number} is odd.")

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

check\_even\_odd(num)

output –



13. Write a program for an iterative factorial of a number.

Source Code:-

def iterative\_factorial(n):

factorial = 1

if n < 0:

return "Factorial is not defined for negative numbers."

elif n == 0:

return 1

else:

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

factorial \*= i

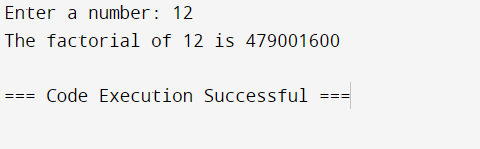
return factorial

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

result = iterative\_factorial(num)

print(f"The factorial of {num} is {result}")

output-



14. write a python program to sort the dictionary by key or a value.

Source Code:-

def sort\_dict\_by\_value(input\_dict):

sorted\_dict = {key: value for key, value in sorted(input\_dict.items(), key=lambda item: item[1])}

return sorted\_dict

input\_dict = {}

n = int(input("Enter the number of key-value pairs: "))

for i in range(n):

key = input(f"Enter key {i+1}: ")

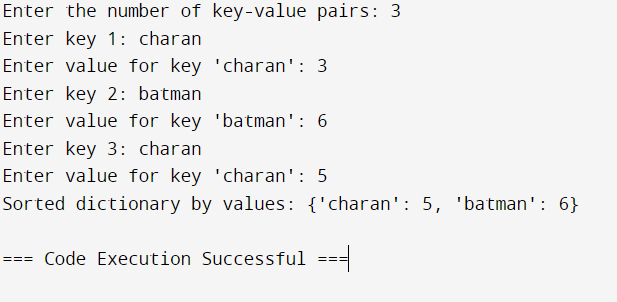
value = int(input(f"Enter value for key '{key}': "))

input\_dict[key] = value

sorted\_dict = sort\_dict\_by\_value(input\_dict)

print("Sorted dictionary by values:", sorted\_dict)

output-



1. Write the python program to find sum of all the items in the dictionary.

Code-

# Dictionary 1

dict\_1 = {'apple': 5, 'banana': 9, 'cherry': 7}

result = sum(dict\_1.values())

print(f"The sum of all values in dictionary 1 is {result}")

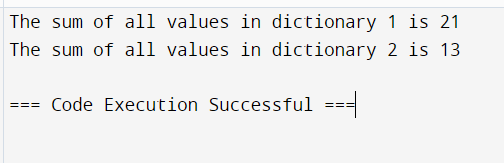
# Dictionary 2

dict\_2 = {'hi': 1, 'hello': 4, 'namaste': 8}

res = sum(dict\_2.values())

print(f"The sum of all values in dictionary 2 is {res}")

output-



1. write a python program to find the size of the dictionary.

Code-

dict\_1 = {'apple': 3, 'banana': 1, 'cherry': 2, 'grape': 4, 'avocado': 5}

result = len(dict\_1)

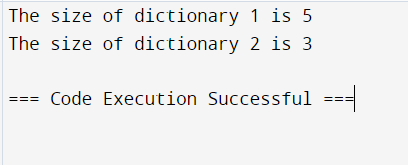
print(f"The size of dictionary 1 is {result}")

dict\_2 = {'hi': 1, 'hello': 4, 'namaste': 8}

res = len(dict\_2)

print(f"The size of dictionary 2 is {res}")

output-



1. write a python program to find the size of a set.

Code-

set\_1 = {6, 7, 8, 9, 10}

result = len(set\_1)

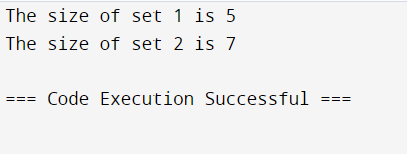
print(f"The size of set 1 is {result}")

set\_2 = {22, 23, 24, 45, 34, 56, 78}

res = len(set\_2)

print(f"The size of set 2 is {res}")

output-



Assignment-3

1. Write a python program for the addition and subtraction of the matrices.

Code-

import numpy

A = numpy.array([[1, 2], [3, 4]])

B = numpy.array([[4, 5], [6, 7]])

print("Printing elements of the first matrix:")

print(A)

print("Printing elements of the second matrix:")

print(B)

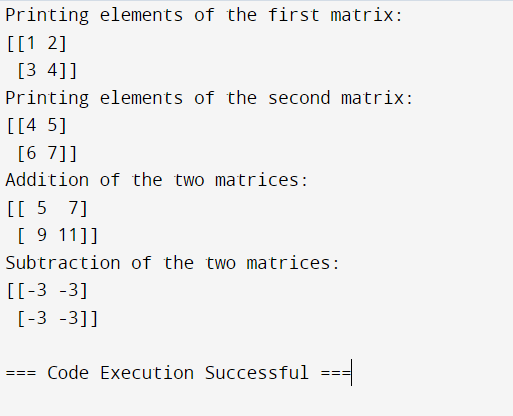
print("Addition of the two matrices:")

print(numpy.add(A, B))

print("Subtraction of the two matrices:")

print(numpy.subtract(A, B))

output-



2) Write a python program for row-wise addition in tuple matrix.

Code-

def row\_wise\_addition(matrix):

num\_rows = len(matrix)

result = []

for i in range(num\_rows):

row\_sum = sum(matrix[i])

result.append(row\_sum)

return result

# Define the matrix with new values

tuple\_matrix = [

(10, 20, 30),

(40, 50, 60),

(70, 80, 90)

]

# Calculate the row-wise addition

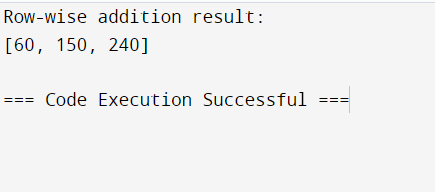
result = row\_wise\_addition(tuple\_matrix)

# Print the result

print("Row-wise addition result:")

print(result)

output-



1. Write a python program to print multiple arguments.

Code-

def print\_arguments(\*args):

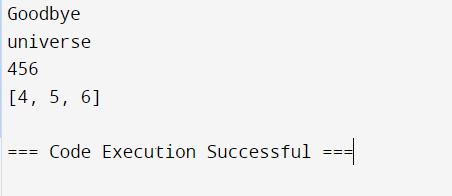
for arg in args:

print(arg)

# Call the function with new values

print\_arguments("Goodbye", "universe", 456, [4, 5, 6])

output-



1. Write a python program to print the power of a number using recursion.

Code-

def power(N, P):

if P == 0:

return 1

return N \* power(N, P - 1)

# Define new values

N, P = 3, 4

print(f"{N} power {P} is {power(N, P)}")

a, b = 2, 5

print(f"{a} power {b} is {power(a, b)}")

output-

