1. **Python Program to print all the Composite numbers in between a and b.**

**Pusedocode:**

1. Start
2. Input by taking two numbers, a and b, where a is the starting number and b in the ending number
3. For each number n from a to b:

If n is less than or equal to 1 : Print as **Invalid**.

1. If n is greater than 1:

For each number i ranges from 2 to (n//2)+1

If n%i==0 then print the values of n

1. Break out the inner loop
2. End.

**INPUT: OUTPUT:**

a=10 Composite numbers= 10 12 14 15 16 18 20

b=20

**PROGRAM:**

a=10

b=20

for n in range(a,b+1):

if n<1:

print("Invalid")

if n>1:

for i in range(2,(n//2)+1):

if n%i==0:

print(n)

break

1. **Python code to find the square and cube of a given decimal number.**

**Pusedocode:**

1. Start
2. Initialize a variable a with the value of 0.6
3. Calculate the square of a:

Square\_num=a\*a

1. Calculate the cube of a:

Cube\_num=sqaure\_num\*a

1. Print the square of a
2. Print the cube of a
3. End

**PROGRAM:**

**a=0.6**

**square\_num=a\*a**

**cube\_num=square\_num\*a**

**print("Sqaure of ",a ,"is:", square\_num)**

**print("cube of ", a, "is:", cube\_num)**

**INPUT: OUTPUT:**

a=0.6 square of number=0.36

Cube of number=0.216

1. **Python code to check whether the given number is palindrome or not.**

**PROGRAM:**

n=121

rev=0

temp=n

while n>0:

r=n%10

rev=rev\* 10 +r

n=n//10

if rev==temp:

print("palindrome")

break

else:

print("Not a palindrome")

**INPUT: OUTPUT:**

n=121 palindrome

1. **Python program to find the Diagonal of the matrix and sum of the diagonal.**

**PROGRAM:**

matrix=[

[1,2,3],

[4,5,6],

[7,8,9],

]

size=len(matrix)

diagonal1=[matrix[i][i] for i in range(size)]

diagonal2=[matrix[i][size-i-1]for i in range(size)]

print("1St diagonal is:", diagonal1)

print("2nd diagoanl is:", diagonal2)

print("Sum of 1st diagonal is:", sum(diagonal1))

print("sum of 2nd diagonal is:", sum(diagonal2))

**INPUT: OUTPUT:**

matrix=[ 1st diagonal=[1, 5, 9]

[1,2,3], 2nd diagoanl=[3, 5, 7]

[4,5,6], sum of 1st diagonal: 15

[7,8,9], sum of 2nd diagonal:15

]

1. **Python program to perform matrix multiplication**

def matrix\_multiply(A, B):

if len(A[0]) != len(B):

return "Matrices cannot be multiplied"

result = [[0 for \_ in range(len(B[0]))] for \_ in range(len(A))]

for i in range(len(A)):

for j in range(len(B[0])):

for k in range(len(B)):

result[i][j] += A[i][k] \* B[k][j]

return result

A = [

[1, 2, 3],

[4, 5, 6]

B = [

[7, 8],

[9, 10],

[11, 12]

]

result = matrix\_multiply(A, B)

for row in result:

print(row)

**INPUT: OUTPUT:**

A = [ [58, 64]

[1, 2, 3], [139, 154]

[4, 5, 6]

]

B = [

[7, 8],

[9, 10],

[11, 12]

]

1. **Python program to return the maximum profit that can be archieved.**

**Program:**

def max\_Profit(price, start, end):

if (end <= start):

return 0;

profit = 0;

for i in range(start, end, 1):

for j in range(i+1, end+1):

if (price[j] > price[i]):

curr\_profit = price[j] - price[i] + maxProfit(price, start, i - 1)+ maxProfit(price, j + 1, end);

profit = max(profit, curr\_profit);

return profit;

if \_name\_ == '\_main\_':

price = [100, 180, 260, 310, 40, 535, 695];

n = len(price);

print(maxProfit(price, 0, n - 1));

**OUTPUT:**

Maximum profit is=100

1. **Python program to find the three numbers from an array such that the sum of three numbers is equal to product of the given number.**

**Program:**

def find\_triplets(array, target\_product):

array.sort()

triplets = []

for i in range(len(array) - 2):

left = i + 1

right = len(array) - 1

while left < right:

current\_sum = array[i] + array[left] + array[right]

current\_product = array[i] \* array[left] \* array[right]

if current\_sum == current\_product and current\_product == target\_product:

triplets.append([array[i], array[left], array[right]])

left += 1

right -= 1

elif current\_sum < target\_product:

left += 1

else:

right -= 1

return triplets

**INPUT: OUTPUT:**

**array = [1, 2, 3, 4, 5, 6, 7, 8, 9]**

**target\_product = 72 [1, 8, 9]**

**[2, 6, 9]**

**[3, 4, 9]**

**[3, 6, 8]**

**[4, 5, 8]**

1. **Python program to find the sum of the given series.**

**Program:**

def factorial(n):

if n == 0:

return 1

else:

return n \* factorial(n-1)

def series\_sum(n):

sum = 0

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

sum += factorial(i) / i

return sum

n = 5

print("Sum of series:", series\_sum(n))

Input: OUTPUT:

1!/1+2!/2+3!/3+4!/4+5!/5 sum=34