1. **Python Program for n-th Fibonacci number**

def Fibonacci(n):

    if n<= 0:

        print("Incorrect input")

    # First Fibonacci number is 0

    elif n == 1:

        return 0

    # Second Fibonacci number is 1

    elif n == 2:

        return 1

    else:

        return Fibonacci(n-1)+Fibonacci(n-2)

# Driver Program

n = int(input("Enter a number: "))

print(Fibonacci(n))

Answer: Enter a number: 7

8

1. **Python Program for How to check if a given number is Fibonacci number?**

num=int(input("Enter the number: "))

temp=0

a=1

b=1

if num==0 or num==1:

    print("Yes")

else:

    while temp<num:

        temp=a+b

        b=a

        a=temp

    if temp==num:

        print("Yes. Number {} is a fibonnaci number".format(num))

    else:

        print("No. Number {} is not a fibonnaci number".format(num))

Answer:

Enter the number: 4

No. Number 4 is not a fibonnaci number

1. **Python Program for n\’th multiple of a number in Fibonacci Series**

def findPosition(k, n):

    f1 = 0

    f2 = 1

    i =2;

    while i!=0:

        f3 = f1 + f2;

        f1 = f2;

        f2 = f3;

        if f2%k == 0:

            return n\*i

        i+=1

    return

# Multiple no.

n = int(input("Enter a Multiple no: "))

# Number of whose multiple we are finding

k = int(input("Enter a Number of whose multiple we are finding: "))

print("Position of {}'th multiple of {} in Fibonacci Series is".format(n, k), findPosition(k,n));

Answer: Enter a Multiple no: 4

Enter a Number of whose multiple we are finding: 5

Position of 4'th multiple of 5 in Fibonacci Series is 20

1. **Program to print ASCII Value of a character**

c = input("Enter the Key: ")

print("The ASCII value of '" + c + "' is", ord(c))

Enter the Key: s

The ASCII value of 's' is 115

1. **Python Program for Sum of squares of first n natural numbers**

def squaresum(n) :

    # Iterate i from 1

    # and n finding

    # square of i and

    # add to sum.

    sum = 0

    for i in range(1, n+1) :

        sum = sum + (i \* i)

    return sum

n = int(input("Enter the number: "))

print("Sum of Square of first {} numbers is: ".format(n),squaresum(n))

Answer:

Enter the number: 10

Sum of Square of first 10 numbers is: 385

1. **Python Program for cube sum of first n natural numbers**

def sumOfSeries(n):

    sum = 0

    for i in range(1, n+1):

        sum +=pow(i,3)

    return sum

n = int(input("Enter the number: "))

print("Sum of cubes of first {} natural numbers is: ".format(n),sumOfSeries(n))

Answer: Enter the number: 10

Sum of cubes of first 10 natural numbers is: 3025

1. **Python Program to find sum of array**

# creating an empty list

arr = []

# number of elements as input

n = int(input("Enter number of elements : "))

# iterating till the range

for i in range(0, n):

    ele = int(input())

    arr.append(ele) # adding the element

ans = sum(arr)

# display sum

print('Sum of the array is ', ans)

Answer: Enter number of elements: 5

1

2

3

4

5

Sum of the array is 15

1. **Python Program to find largest element in an array**

# creating an empty list

arr = []

# number of elements as input

n = int(input("Enter number of elements : "))

# iterating till the range

for i in range(0, n):

    ele = int(input())

    arr.append(ele) # adding the element

# in arr[] of size n

def largest(arr, n):

    ans = max(arr)

    return ans;

# Driver code

if \_\_name\_\_ == '\_\_main\_\_':

       print ("Largest in given array ", largest(arr, n))

Answer: Enter number of elements : 3

120

234

32

Largest in given array 234

1. **Python Program for array rotation**

# creating an empty list

arr = []

# number of elements as input

n = int(input("Enter number of elements : "))

# iterating till the range

for i in range(0, n):

    ele = int(input())

    arr.append(ele) # adding the element

# slicing approach to rotate the array

def rotateList(arr,d,n):

  arr[:]=arr[d:n]+arr[0:d]

  return arr

# Driver function to test above function

print(arr)

print("Rotated list is")

print(rotateList(arr,n-1,len(arr)))

Answer: Enter number of elements : 5

5

4

3

2

1

[5, 4, 3, 2, 1]

Rotated list is

[4, 3, 2, 1, 5]

1. **Python Program for Reversal algorithm for array rotation**

# creating an empty list

arr = []

# number of elements as input

n = int(input("Enter number of elements : "))

r = int(input("Rotate elements by number: "))

# iterating till the range

for i in range(0, n):

    ele = int(input())

    arr.append(ele) # adding the element

# slicing approach to rotate the array

def rotateList(arr,d,n):

  arr[:]=arr[d:n]+arr[0:d]

  return arr

# Driver function to test above function

print(arr)

print("Rotated array is")

print(rotateList(arr,r,len(arr)))

Answer: Enter number of elements : 5

Rotate elements by number: 4

5

4

3

2

1

[5, 4, 3, 2, 1]

Rotated array is

[1, 5, 4, 3, 2]

1. **Python Program to Split the array and add the first part to the end**

# creating an empty list

arr = []

# number of elements as input

n = int(input("Enter number of elements : "))

r = int(input("Position elements need to be added: "))

# iterating till the range

for i in range(0, n):

    ele = int(input())

    arr.append(ele) # adding the element

print(arr)

x = arr[:r]

y = arr[r:]

y.extend(x)

for i in y:

    print(i, end=" ")

**Answer:** Enter number of elements : 5

Position elements need to be added: 4

5

4

3

2

1

[5, 4, 3, 2, 1]

1 5 4 3 2