Date: 28/12/2020

**PRACTICAL 1**

# AIM: Develop Programs to Understand the Control Structures, Branching Programs, Strings and Input of Python and functions.

**PRACTICAL 1.1**

# AIM: Write a Python Program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700.

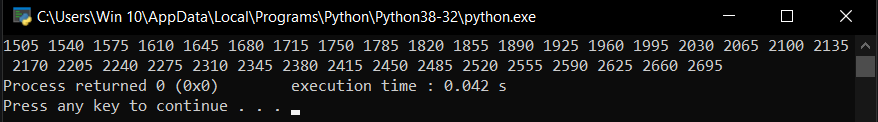
**PROGRAM:**

for i in range(1500, 2701, 5):

if i % 7 == 0:

print(i, end=" ")

**OUTPUT:**



**PRACTICAL 1.2**

# AIM: Write a Python program to construct the following pattern, using nested for loop.

# \*

# \* \*

# \* \* \*

# \* \* \* \*

# \* \* \* \* \*

# \* \* \* \*

# \* \* \*

# \* \*

# \*

**PROGRAM:**

def pattern(N):

# forward triangle

for i in range(0,N):

for j in range(0, i+1):

print("\* ",end="")

print()

# backward triangle

for i in range(1, N):

for j in range(N, i, -1):

print("\* ", end="")

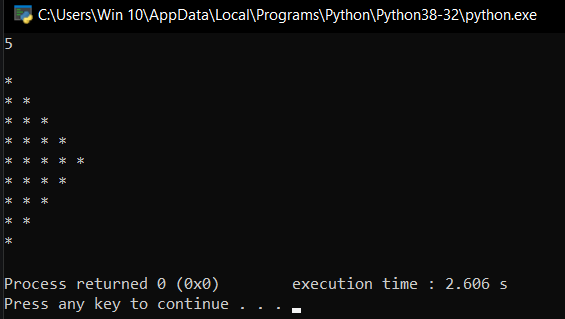
print()

N = int(input())

print()

pattern(N)

**OUTPUT:**



**PRACTICAL 1.3**

# AIM: Write a Python program that accepts a word from user and reverse it (without using the reverse function)

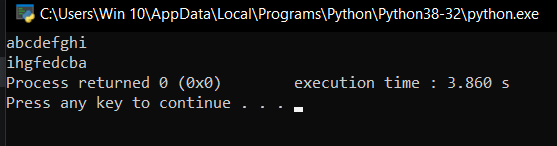
**PROGRAM:**

string = input()

for x in range(len(string)-1, -1, -1):

print(string[x], end="")

**OUTPUT:**



**PRACTICAL 1.4**

# AIM: Write a Python program to check whether an alphabet is a vowel or consonant.

**PROGRAM:**

def check(char):

vowels = ['a', 'e', 'i', 'o', 'u']

if char in vowels:

print("It is Vowel")

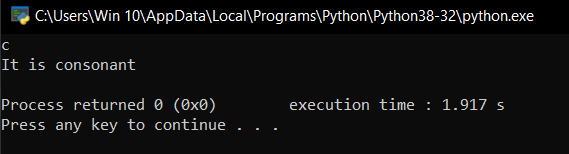
else:

print("It is consonant")

char = input()

check(char)

**OUTPUT:**



**PRACTICAL 1.5**

# AIM: Write a Python program to find reverse of given number using user defined function.

**PROGRAM:**

def reverse(var):

r = 0

while var > 0:

d = var % 10

r = (r \* 10) + d

var = var // 10

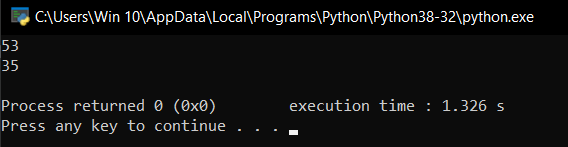
return r

N = int(input())

rev = reverse(N)

print(rev)

**OUTPUT:**



**PRACTICAL 1.6**

# AIM: Write a Python program to check whether the given no is Armstrong or not using user defined function.

**PROGRAM:**

def armstrong(N):

length = len(N)

arm = 0

for i in N:

arm = arm + (int(i) \*\* length)

if arm == int(N):

print("It is Armstrong!!")

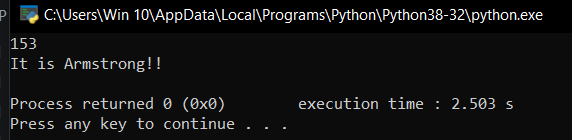
else:

print('It is not Armstrong!!')

N = input()

armstrong(N)

**OUTPUT:**



**PRACTICAL 1.7**

# AIM: To write a Python program to find first n prime numbers.

**PROGRAM:**

def printprime(n):

num = n - 1

n = 3

print("2 ", end="")

# counts till first n

while num > 0:

# checks each prime number

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

if n % i == 0:

n += 1

break

else:

print(n, end=" ")

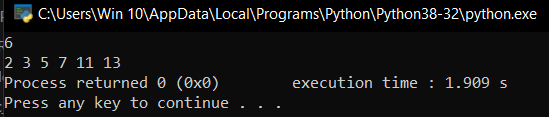
num -= 1

n += 1

N = int(input())

printprime(N)

**OUTPUT:**



**PRACTICAL 1.8**

# AIM: Write a Python program to print Fibonacci series upto n terms.

**PROGRAM:**

def fibo(n):

a = 0

b = 1

print(a, b, end=" ")

for \_ in range(2, n):

c = a + b

print(c, end=" ")

a = b

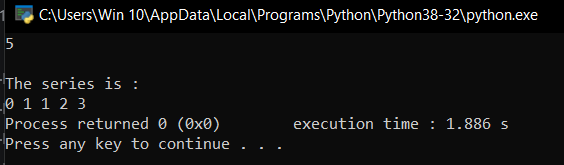
b = c

N = int(input())

print("\nThe series is : ")

fibo(N)

**OUTPUT:**



**PRACTICAL 1.9**

# AIM: Give the output of following Python code:

**a)** myStr = ‘GTU is the best University’

print myStr [15 : : 1]

print myStr [-10 : -1 : 2]

**OUTPUT:**

University

Uiest

**b)** t = (1, 2, 3, (4, ), [ 5, 6] )

print t[ 3 ]

t[4][0] = 7

print t

**OUTPUT:**

(4,)

(1, 2, 3, (4,), [7, 6])

**c)** I=[(x, y) for x in [1,2,3]

for y in [3,1,4] if x != y]:

print I

**OUTPUT:**

[(1, 3), (1, 4), (2, 3), (2, 1), (2, 4), (3, 1), (3, 4)]

**d)** str1 = ‘This is Pyhton’

print "Slice of String : ", str1[1 : 4 : 1]

print "Slice of String : ", str1[0 : -1 : 2]

**OUTPUT:**

Slice of String : his

Slice of String : Ti sPho