

# Ex 4: Python Programming using Loops

**Aim**

# TO EXECUTE PYTHON PROGRAMS USING LOOPS AND PRINT THE OUTPUT OF THE FOLLOWING PROGRAMS.

1 . Compute Exponentiation (power of a number) without using

\*\* operator. Python Code:

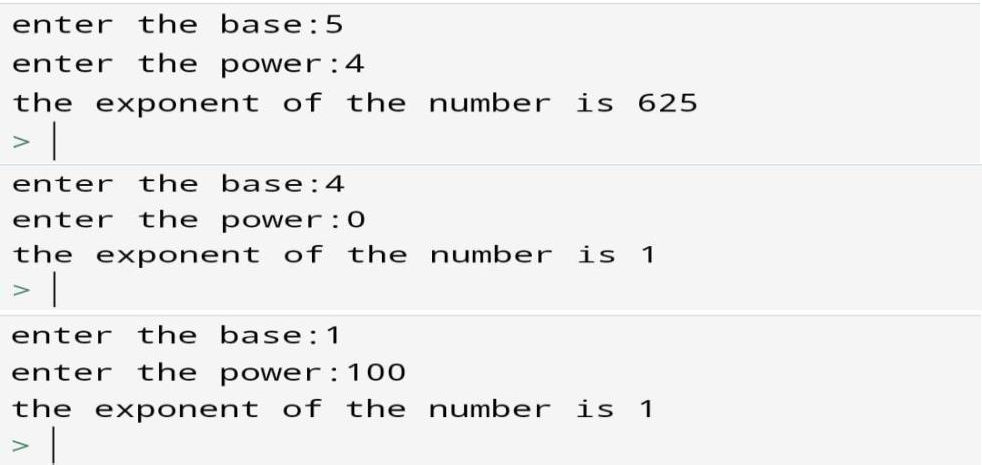
#display power of a number without using \*\* operator x=int(input("Enter the no. = "))

y=int(input("Enter the power to be raised = ")) c,p=1,1

while c<=y: c+=1

p\*=x print("Answer = ",p)

Test Cases:





Q.No 2 : Print the sum of all the digits of a number.

PYTHON CODE

#print the sum of all digits in a number n=int(input("enter a no: "))

s=0

while n!=0 :

d=n%10 s+=d n=n//10

print(s)

TEST CASES





Q.No 3 : Print all the two digit numbers which are either divisible by 3 or by 4.

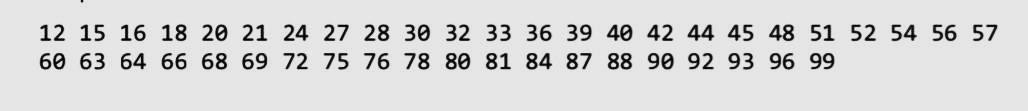
PYTHON CODE

Print 2 digit numbers divisible by 3 or 4 n=10

while(n<100):

if (n%3==0 or n%4==0): print (n)

n+=1 TEST CASES





Q.No 4 : Perform the division operation and find the quotient and remainder values. (Without using /, // % operators)

PYTHON CODE

#Perform division operation without using /,//,% dividend=int(input("Enter dividend : ")) divisor=int(input("Enter divisor :"))

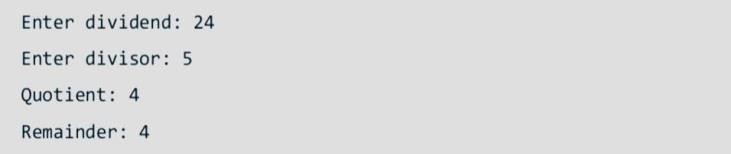
c,sum =0,0

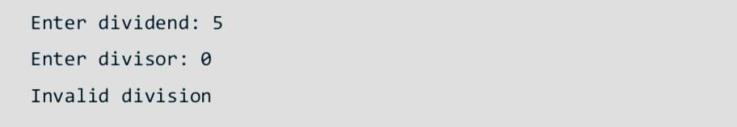
while((dividend-sum)>=divisor):

sum+=divisor c+=1

remainder=dividend-sum print("Quotient: ",c) print("Remainder",remainder)

TEST CASES







Q.No 5 : Check whether the given number is palindrome or not.

PYTHON CODE

#Check whether the given no. is palindrome or not n=int(input("Enter a no:"))

t=n

r=0 while(t!=0):

d=t%10 r=r\*10+d t=t//10

if(n==r):

print("the given number is palindrome ") else :

print("the given number is not a palindrome") TEST CASES







6 : Check whether the entered number is an Armstrong number or not.

For example: 153= 13 + 53 + 33 = 153 is an Armstrong number. PYTHON CODE

#Check whether the no. is Armstrong no. or not n=int(input("Enter a no:"))

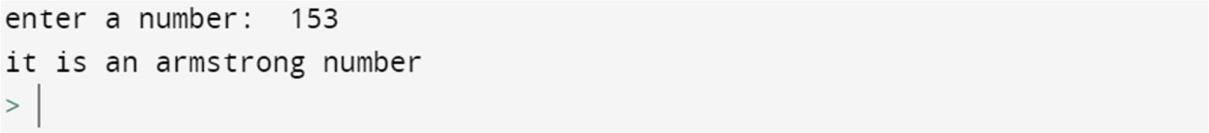
t,s=n,0 while n!=0 :

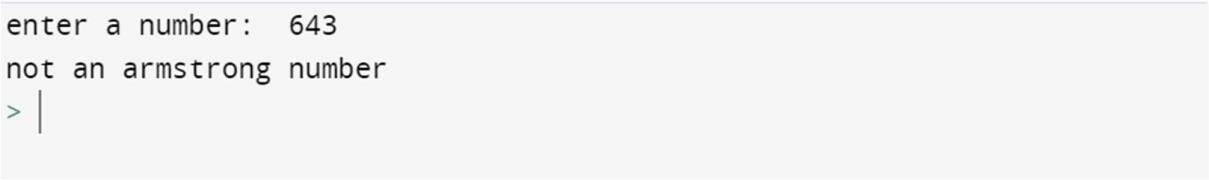
d=n%10 s+=d\*\*3 n=n//10

if(s==t):

print("it is a armstrong number") else:

print("it is not a armstrong number") TEST CASES







Q.No 7 : Compute the GCD of two numbers.

PYTHON CODE

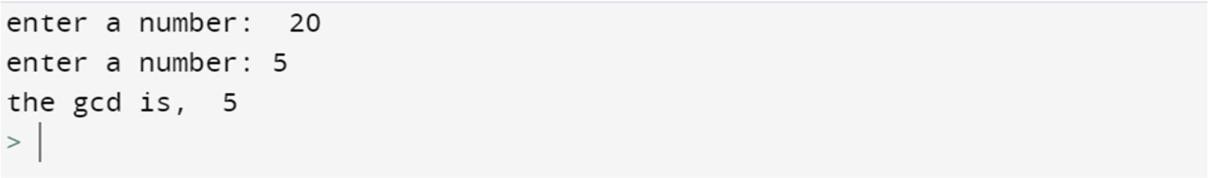
#Compute GCD of two nos. x=int(input("enter a number : ")) y=int(input("enter a number :")) if(x>y):

x,y=y,x while(y%x!=0):

t=y y=x x=t%x

print("GCD = ",x)

TEST CASES





Q.No 8 : Write a python program for the below mentioned scenario.Program will require a input in voltage if it is 5 V, then it is active if less than 5V, it is CUTOFF ,if greater than 5V it is Breakdown .Your program will continue to ask for input voltage until you enter a CUTOFF VOLTAGE. In addition, the program will terminate if there is a Breakdown voltage as input.

PYTHON CODE

#voltage program while(1):

volt = int(input("enter the voltage:")) if(volt<5):

print("CUTOFF") break

elif(volt>5):

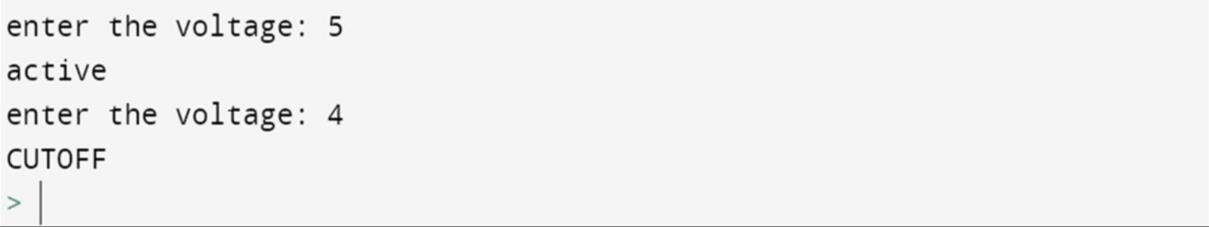
print("BREAKDOWN")

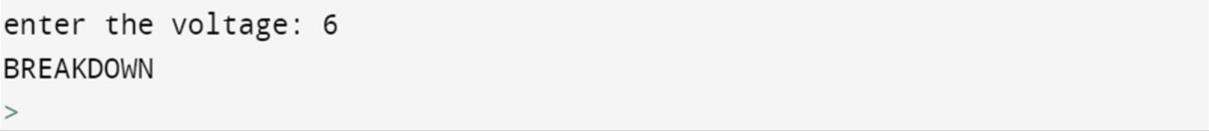
break

else:

print("active")

TEST CASES







Q.No 9 : Write a program to find the square root of a number

using Newton’s method

PYTHON CODE

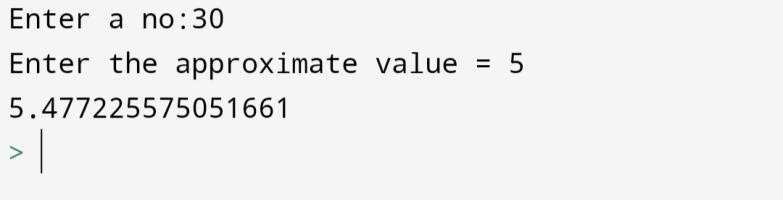
#Find the sqaure root of a no. using newtons method n=int(input("Enter a no:"))

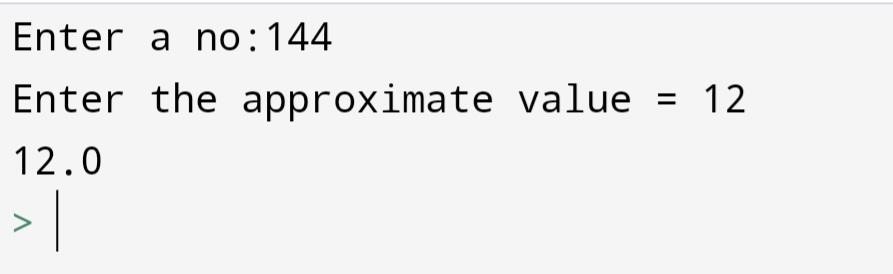
aprox=int(input("Enter the approximate value = ")) sqrt=1

while sqrt!=aprox: aprox = sqrt

sqrt = (aprox +(n/aprox))/2 print(sqrt)

TEST CASE







Q.No 10 : Print the nth multiplication table.

PYTHON CODE

#Print the nth multiplication table n=int(input("Enter any number:"))

print("The multiplication table of ",n," : ") for i in range(1,21):

mult=n\*i print(n,"\*",i,"=",mult)

TEST CASES





Q.No 11 : Print all the prime numbers between 1 to 100.

PYTHON CODE

#print all the prime nos. between 1 to 100 def prime(x):

c=0

for i in range(1,x+1): if(x%i==0):

c+=1

if c==2 :

return True return False

print("Prime nunmers between 1 to 100 : ") for i in range(2,100):

if prime(i): print(i,end=" ")

print() TEST CASES





Q.No 12 : Find the factorial of a number.

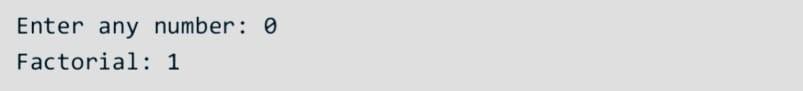
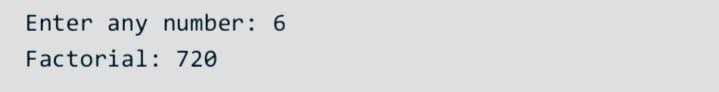
PYTHON CODE

#Find the factorial of a number a=int(input("Enter any number: ")) f=1

for i in range(1,a+1): f\*=i

print("Factorial :",f)

TEST CASES



Q.No 13 : Write python programs to print the following sequences,

a. 1,3,9,27,81,243...

b. -4, -4, -,2,4

c. 1,8,27,64... PYTHON CODE

#Print the following sequences

#1,3,9,27,81,243,...

n1=int(input("Enter the no. of terms : ")) a=1

for i1 in range(n1): if i1!=n1-1:

print(a,end=",")

else:

a\*=3 print()

print(a,end="")

#-4,-2,0,2,4

n2=int(input("Enter the no. of terms : ")) b=-4

for i2 in range(n2): if i2!=n2-1:

print(b,end=",") else:

print(b,end="")

b+=2

print()

#1,8,27,64

n3=int(input("Enter length of : ")) for i3 in range(1,n3+1):

if i3!=n3:

print(i3\*\*3,end=",") else:

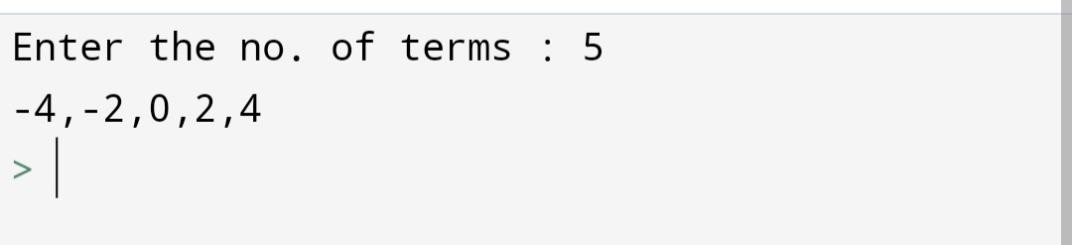
print(i3\*\*3,end="")

print()



TEST CASES







14 : Find the sum of series:

a. 1 + x^2/2 + x^3/3 + ... x^n/n

b. -x + x2 – x3 +x4+....

PYTHON CODE

#Find the sum of the series #a

n1=int(input("Enter any number: "))

x1=int(input("Enter length of the series : ")) s1=0

for i1 in range(2,n1+1): s1=s1+((x1\*\*i1)/i1)

s1=s1+1

print("Sum =",s1)

#b

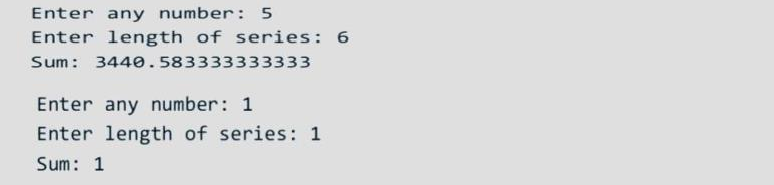
n2=int(input("Enter any number:")) x2=int(input("Enter length of series:")) s2=0

for i2 in range(0,n2): s2+=((x2\*\*(i2+1))\*((-1)\*\*(i2+1)))

print(((x2\*\*(i2+1))\*((-1)\*\*(i2+1))),end=" ") print("Sum =",s2)

TEST CASES

a



b







Q.No 15 : Print the Fibonacci series up to N numbers. Ex: Fibonacci Series = 0 1 1 2 3 5 8 13 21....

PYTHON CODE

#print the fibonacci series up to N a=0

b=1

N=int(input("Enter the length of series : ")) if N==1:

print(a) elif N==2:

print(a,b) else:

print(a,b,end=" ") for i in range(2,N):

t=a+b print(t,end=" ") a=b

b=t

print()

TEST CASES





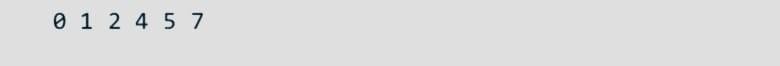
Q.No 16 : Write a Python program that prints all the numbers from 0 to 7 except 3 and 6. Note: Use 'continue' statement.

PYTHON CODE

#Print all the numbers from 0 to 7 except 3 and 6 for i in range(0,8):

if(i==3 or i==6): continue print(i,end=" ")

print() TESTCASES



Q.No 17 : Print the following patterns:



Python code:

#program to print the pattern #a

for i in range(1,5):

for j in range(i,0,-1): print(j,end=" ")

print()

#b

for i in range(5):

print(" "\*(5-i-1),end="") for j in range(i+1):

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

#c

from math import factorial n = 7

for i in range(n):

for j in range(n-i+1): print(end=" ")

for j in range(i+1):

res = factorial(i)//(factorial(j)\*factorial(i-j)) tempres=res

count = 0 while(tempres>0):

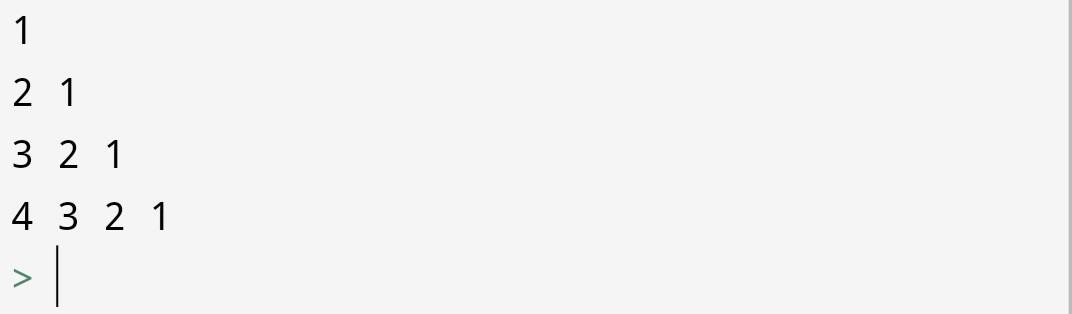
count+=1 tempres=tempres//10

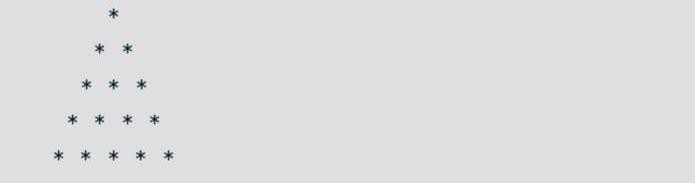
if(count==2):

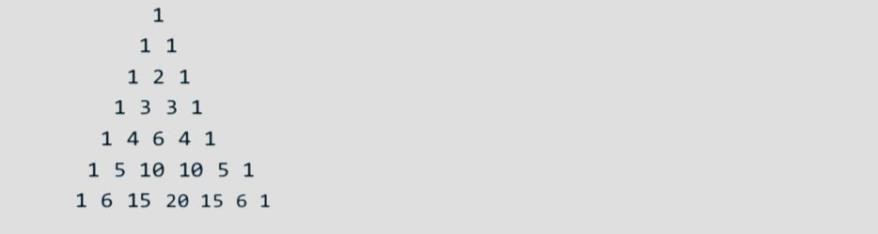
print(res," ",sep="",end="") elif(count==1):

print(res," ",sep="",end="")

print() TEST CASES







# PART – B(OPTIONAL)



Q.NO 1: (Financial application: compute future tuition) Suppose that the tuition for a university is $10,000 this year and increases 5% every year. Write a program that computes the tuition in ten years and the total cost of four years’ worth of tuition starting ten years from now.

PYTHON CODE

fee = 10000 rate = 5/100 count = 0

tot = 0 while(count<=14):

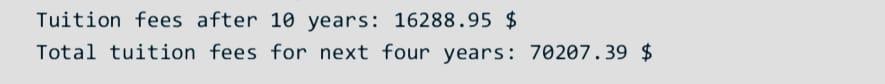
count+=1

fee = fee + (fee\*rate) if(count==10):

print("Tuition fees after 10 years:", fee) if(count>10):

tot+=fee

print("Total tuition fees for next four years:",fee) TEST CASES





Q.NO 2: (Find the highest score) Write a program that prompts the user to enter the number of students and each student’s score and displays the highest score and second highest score.

PYTHON CODE

n = int(input("Enter total number of students:")) high = 0

secondhigh = 0

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

pmt = "Enter student {} score:".format(i) val = int(input(pmt))

if(val>high):

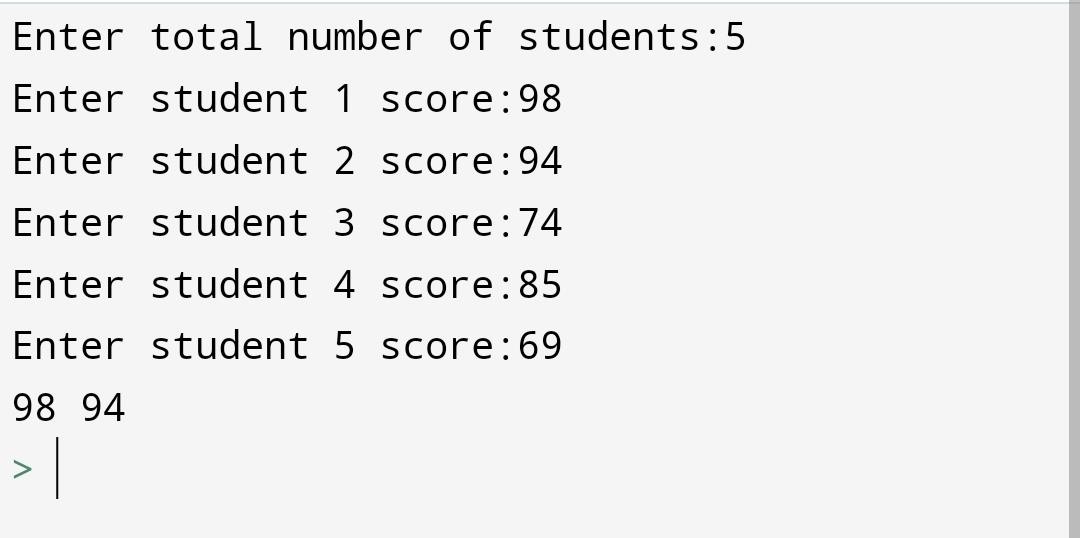
if(secondhigh!=0 and high!=0): secondhigh,high = high,val

else:

high = val if(val>secondhigh and val<high):

secondhigh = val print(high,secondhigh)

TEST CASES





Q.NO 3: Write a program to convert a binary number to decimal number and vice versa using loops.

PYTHON CODE

#decimal to binary

a=int(input('Enter a decimal number to be converted to binary:'))

r=0 c=0

while a>0:

r+=10\*\*c\*(a%2) c+=1

a//=2 print(r)

#binary to decimal

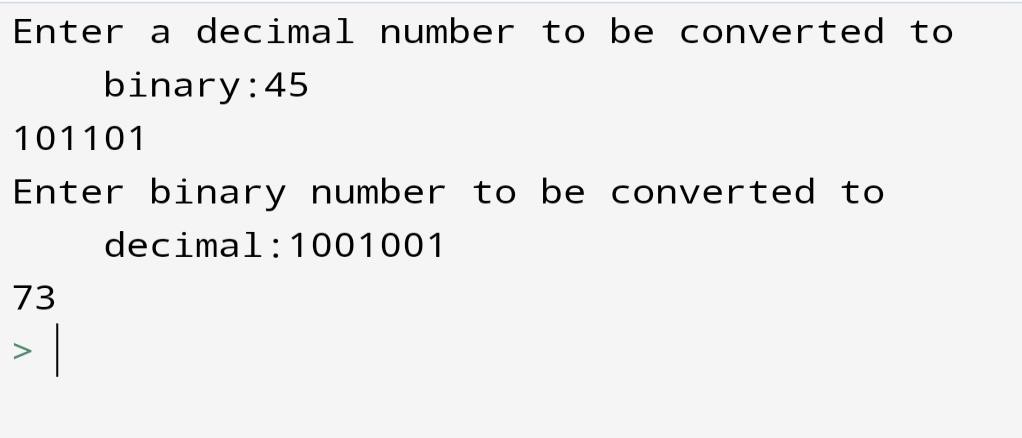
a=int(input('Enter binary number to be converted to decimal:'))

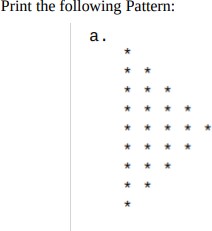
s,c=0,0

while a>0: s+=2\*\*c\*(a%10) c+=1

a//=10 print(s)

TEST CASES



Q.NO 4:

PYTHON CODE

# right pascal triangle n = 5

# upper triangle for i in range(n):

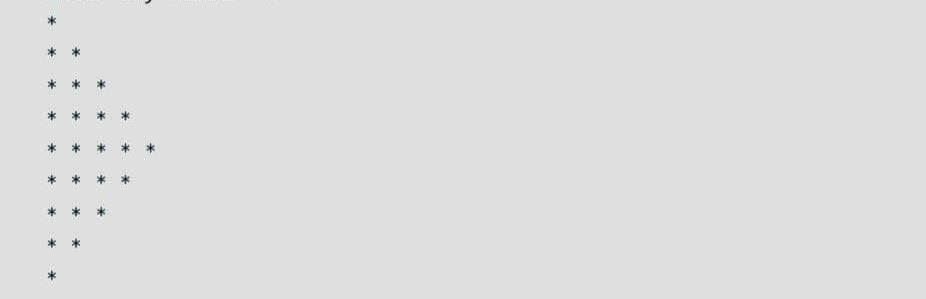
for j in range(i + 1): print('\*', end="")

print()

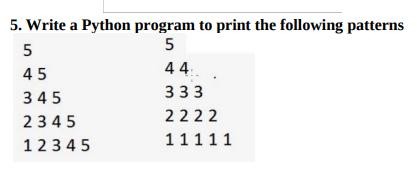
# lower triangle for i in range(n):

for j in range(n - i - 1): print('\*', end="")

print() TEST CASES



Q.NO 5:



PYTHON CODE

#program to print the first pattern n=int(input("Enter any number:")) for i in range (n):

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

print(n-j,end=" ")

print()

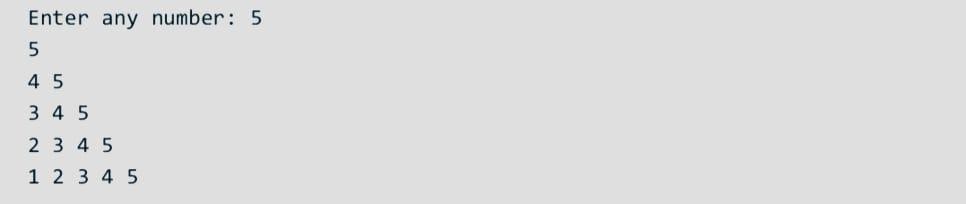
#program to print the second pattern n=int(input("Enter any number:")) for i in range (n):

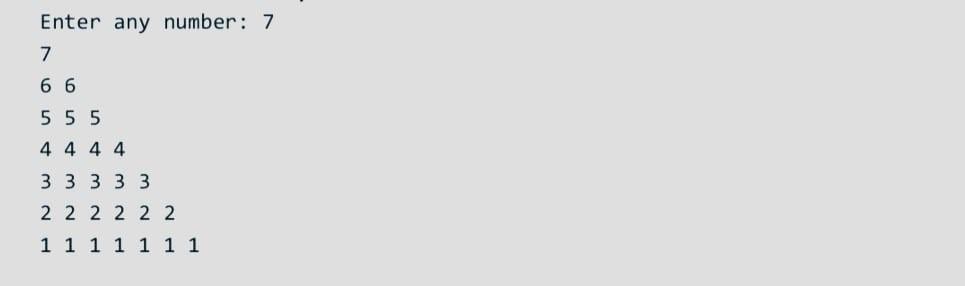
for j in range(i+1): print(n-i,end=" ")

print()

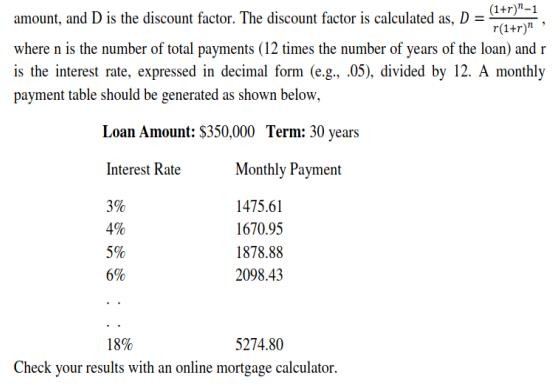


TEST CASES





6: Develop and test a Python program that calculates the monthly mortgage payments for a given loan amount, term (number of years) and range of interest rates from 3% to 18%. The fundamental formula for determining this is A/D, where A is the original loan



Python code:

#mortgage table A=int(input('Enter loan amount:'))

t=int(input('Enter number of years:'))

R=3

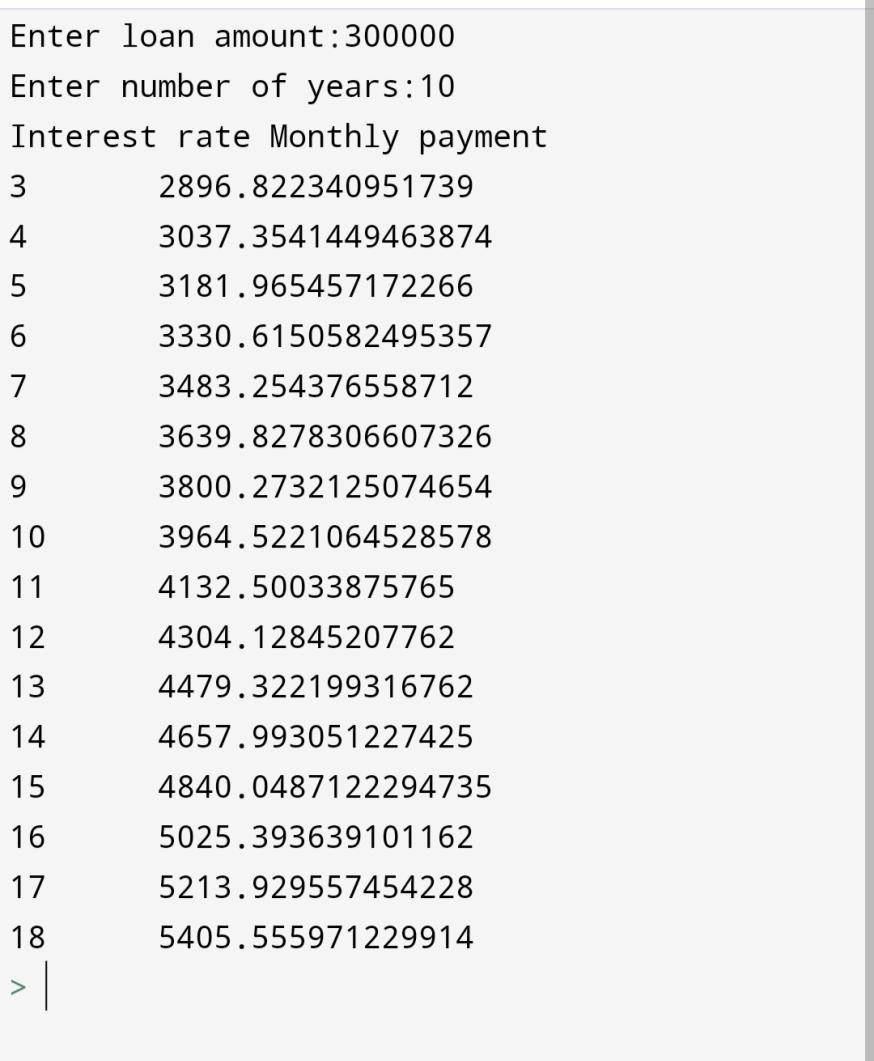
print('Interest rate','Monthly payment') while R<=18:

r=(R/100)/12

n=t\*12

D=((1+r)\*\*n -1)/(r\*(1+r)\*\*n) print(R,A/D,sep=2\*'\t') R+=1





**Learning Outcomes:**

**I have learnt and developed programs:**

1. To use "while" loop in different scenarios.
2. To use "for" loop for different logical questions and different patterns
3. To use multiple, nested looping structures.