

11. Strong number or not

q=n

```
while(q>0):           #line 3
    r=q%10
    fact=1
    for i in range(1,r+1):   #line 6
        fact=fact*i
    sum1=sum1+fact
    q=q//10
if(sum1==n):           #line 10
    print("Strong number it is ")   #line11
else:
    print("It is not Strong number") #line13
```

Input: q=145

while(145>0):	while(14>0):
r=145%10=5	r=14%10=4
fact=1	fact=1
for i in range(1,5+1):	range(1,4+1):
fact=fact*i(1*2*3*4*5)=120	fact=24
sum1=0+120=120	sum1=120+24
q=145//10=14	= 144
	q=14//10=1
	Similar to q=1

Finally output sum1=145 , print output

if(sum1==n): At line11

12. Armstrong number or not

q=n

```
while(q>0):
    r=q%10
    sum1=sum1+r*r*r
    q=q//10
if(sum1==n):
    print(" It is Armstrong number") #line9
else:
    print(" It is not Armstrong number") #line11
```

n=145

while(371>0) :	while(37>0):	(3>0)
r=371%10=1	r=37%10=7	3%10=3
sum1=0+1*1*1=1	sum1=1+7*7*7=344	344+27
q=371//10=37	q=37//10=3	=371
		q=3//10
		q=0

Finally we get the sum1=371

If(sum1==n):

Execute **LINE11**

ELSE:


Execute **LINE13**

13. Fibonacci Series

n=int(input("enter n value:"))	n=6			
a=1	a=1			
b=1	b=1			
print(a,b,end=" ")	It print 1 1			
for i in range(2,n):	for i in range(2,6):	range(2,6):	range(2,6):	range(2,6):
res=a+b	res=1+1=2	res=1+2=3	res=2+3=5	res=3+5=8
a=b	a=1	a=2	a=3	a=5
b=res	b=2	b=3	b=5	b=8
print(res,end=" ")	It print 2	Print 3	print 5	print 8

The Final Output will be like 1 1 2 3 5 8 when (n=6)

13(a).Sum of the Fibonacci series

n=int(input("enter number:"))	n=4		
def calsum(n):	calsum(4):		
fib=[0]*(n+1)	fib=[0,0,0,0,0]		
fib[0]=0	fib[0]=0		
fib[1]=1	fib[1]=1		
sm=fib[0]+fib[1]	sm=0+1=1		
for i in range(2,n+1):	range(2,4+1):	range(2,5):	range(2,5):
fib[i]=fib[i-2]+fib[i-1]	fib[2]=fib[0]+fib[1]=1	fib[3]=fib[1]+fib[2]	fib[4]=fib[2]+fib[3]
sm=sm+fib[i]	sm=1+1=2	 => 1+1=2	=>1+2=3
return sm	return 2	sm=2+2=4	sm=4+3=7
calsum(n)		return 4	return 7

1.Input:n=3

Output:4

Explanation:0+1+1+2=4

2.Input:n=4

Output:7

Explanation:0+1+1+2+3=7

14. Factorial Using Recursion

n=int(input("enter number:"))	n=5	
def fact(n):	fact(5):	fact(4):
if n==1:	if 1==1:	if 1==1:
return 1	return 1	return 1
elif n>1:	elif 5>1:	elif 4>1:
return n*fact(n-1)	return 5*fact(4)	return 4*fact(3)
fact(n)		

Similarly do the fact(3),fact(2),fact(1)

Where fact(1)=1

So the Output is $5*4*3*2*1=120$

15. Power of the given number

nbr=int(input("enter number:"))	n=2	
base=int(input("enter base:"))	b=5	
temp=1	temp=1	
for i in range(1,nbr+1):	range(1,2+1):	range(1,3):
temp=temp*base	temp=1*5=5	temp=5*5=25
print(temp,end=" ")		It print temp as 25 and program terminates

16. Perfect number or not

n=int(input("enter number:"))	n=6			
sum1=0	sum1=0			
q=n	q=6			
for i in range(1,q):	range(1,6):	range(1,6):	range(1,6):	range(1,6):
if q%i==0:	if 6%1==0:	if 6%2==0:	if 6%3==0:	if 6%4==0:
sum1=sum1+i	sum1=0+1=1	sum1=1+2=3	sum1=3+3=6	Here 6 is not
if(sum1==n):				divisible by 4
print("It is perfect number") #line8				&6%5!=0
else:				So the Sum1=6
print("It is not perfect number")				

17.Gcd of two numbers using recursion

n1=int(input("enter n1:"))	n1=4			
n2=int(input("enter n2:"))	n2=6			
def gcd(n1,n2):	gcd(4,6):	gcd(6,4):	gcd(4,2):	gcd(2,0):
if n2!=0:	if 6!=0:	if 4!=0:	if 2!=0:	if 0!=0:
return gcd(n2,n1%n2)	gcd(6,4%6)	gcd(4,6%4)	gcd(2,4%2)	It is false
else:	=>gcd(6,4)	=>gcd(4,2)	=>gcd(2,0)	So it goes to else part and
return n1				Print n1(2)
gcd(n1,n2)				

17(a).Gcd of two numbers

n1=int(input("enter n1:"))	n1=4		
n2=int(input("enter n2:"))	n2=6		
while(n1>0) and(n2>0):	(4>0) and (6>0):	(4>0) and(2>0):	(2>0) and(2>0):
if(n1>n2):	if 4>6:	4>2:	2>2:
n1=n1-n2		n1=4-2=2	False So it goes out of loop
else:	else:		Print(2)
n2=n2-n1	n2=6-4=2		
print(n1,end=" ")			

Special type of programs from (18-21)¶

18. Automorphic number¶

An Automorphic number is that number whose square ends with the same digits as the original number

Ex: $5^2=25$

$$6^2=36$$

$$76^2=5776$$

$$376^2=141376$$

n=int(input("enter number:"))	n=25		
def calsum(n):	calsum(25):		
sum1=0	sum1=0		
sq=n*n	sq=25*25=625		
while(n>0):	while(25>0):	while(2>0):	
if(n%10!=sq%10):	if (25%10!=625%10):	if(2%10!=62%10):	
return 0	return 0	return 0	
n=n//10	n=25//10=2	n=2//10=0	
sq=sq//10	sq=625//10=62	sq=62//10=6	
return 1			In next step n=0 so it terminates print
if(calsum(n)):		return 1	
print("It is automorphic")			
else:			
print("It is not automorphic")			

19.Harshad number or not

n=int(input("enter number:"))	n=153			
sum1=0	sum1=0			
q=n	q=153			Total sum1=9
while(q>0):	while(153>0):	while(15>0):	while(1>0):	if(153%sum1==0)
r=q%10	r=153%10=3	r=15%10=5	r=1%10=1	It is Harshad
sum1=sum1+r	sum1=0+3=3	sum1=3+5=8	sum1=8+1=9	
q=q//10	q=153//10=15	q=15//10=5	q=1//10=0	
if(n%sum1==0):				
print("It is Harshad number")				
else:				
print("It is not Harshad number")				

20. Abundant number or not

n=int(input("enter number:"))	n=18	range(1,18):
sum1=0	sum1=0	if 18%4==0:
for i in range(1,n):	for i in range(1,18):	It is not possible
if n%i==0:	if 18%1==0:	range(1,18):
sum1=sum1+i	sum1=0+1=1	if 18%5==0:
	range(1,18):	It is not possible
if(sum1>n):	if 18%2==0:	similarly so on upto 17
print("It is abundant")	sum1=1+2=3	sum1=1+2+3+6+9=21
else:		if(21>18):
print("It is not abundant")		It is Abundant

21. Amicable or Friendly program

n1=int(input("enter number1:"))	n=220	
n2=int(input("enter number2:"))	n2=284	
sum1=0	sum1=0	
sum2=0	sum2=0	
for i in range(1,n1):	range(1,220):	range(1,284):
if n1%i==0:	Therefore factors for	Therefore factors for 284
sum1=sum1+i	220 are 1,2,4,5,10,22,44,55,110	is 1,2,4,71,142
	Their sum is sum1=284	Their sum is sum2=220
for j in range(1,n2):		
if n2%j==0:		
sum2=sum2+j		
if(sum1==n2)and(sum2==n1):		
print("Amicable number")		
else:		
print("It is not Amicable number")		