Write a Python program to find the sum of the digits of a given number.

In mathematics, the factorial of a positive integer n, denoted by n!, is the product of all positive integers less than or equal to n

In [ ]:
```
def fact(n):
    f=1
    for i in range(1,n+1): #calculating the factorial through a simple for loop
        f*=i
    return f

def sum(a): #sum calculated by using modulo operator and then adding each remain
    b=0
    while(a>0):
        b+=(a%10)
        a=a//10
    return b



n=int(input("Enter the number: ")) #input from the user and typecasting it

fac=fact(n) #function call for factrial
s=sum(fac) #function call for sum of the numbers in factorial
#output
print(f"Factorial of {n} is {fac}")
print(f"Sum of numbers in the factorial {fac} is {s}")
```

```
Factorial of 5 is 120
Sum of numbers in the factorial 120 is 3
```

Use bitwise operators to check if a given number is a power of 2. Write a function that returns True if the number is a power of 2, otherwise False.

In [ ]:
```
def pwr(n):
    if(n<0): #checking whether zero or not
        return False
    return (n&(n-1)==0) #performing AND operation on the number and the previous
    #result of the bitwise AND operation will be zero


n=int(input("Enter the number: ")) #input from user and typecasstong to int

if(pwr): #function call
    print(f"Given number {n} is a power of 2")
else:
    print(f"Given number {n} is not a power of 2")
```

```
Given number 256 is a power of 2
```

Write a Python program to find the sum of all the positive integers which cannot be written as the sum of two abundant numbers.

In number theory, an abundant number is a number for which the sum of its proper divisors is greater than the number itself.

In [ ]:

```
#
def sum(n): #taking the sum of the divisors
    s=1
    i=2
    while i*i<=n:
        if n%i==0:
            s=s+i
            if i!=n//i:
                s=s+(n//i)
        i=i+1
    return s

def abdt(n): #check if abundant or not
    return sum(n)>n

LIMIT = 28123 #setting upper bound

abd=[]
#generating all abundant numbers upto the limit
for i in range(12,LIMIT+1):
    if abdt(i):
        abd.append(i)

w=[False]*(LIMIT+1) #boolean array to mark sum of two abundant number

for i in range(len(abd)):
    for j in range(i,len(abd)):
        s=abd[i]+abd[j]
        if s <= LIMIT:
            w[s]=True
        else:
            break

total=0
#sum of all the numbers that cannot be written as the sum of two abundant number
for i in range(1, LIMIT + 1):
    if not w[i]:
        total=total+i

print("Sum of all positive integers which cannot be written")
print("as the sum of two abundant numbers:", total)
```

```
Sum of all positive integers which cannot be written
as the sum of two abundant numbers: 4179871
```