Question1. Create a function that takes three arguments a, b, c and returns the sum of the numbers that are evenly divided by c from the range a, b inclusive.

**Examples**

evenly\_divisible(1, 10, 20) ➞ 0

# No number between 1 and 10 can be evenly divided by 20.

evenly\_divisible(1, 10, 2) ➞ 30

# 2 + 4 + 6 + 8 + 10 = 30

evenly\_divisible(1, 10, 3) ➞ 18

# 3 + 6 + 9 = 18

def evenly\_divisible(a,b,c):

sum=0

for i in range(a,b+1):

if (i%c==0) :

sum+=i

return sum

print(evenly\_divisible(6,15,3))

Question2. Create a function that returns True if a given inequality expression is correct and False otherwise.

### Examples

correct\_signs("3 < 7 < 11") ➞ True

correct\_signs("13 > 44 > 33 > 1") ➞ False

correct\_signs("1 < 2 < 6 < 9 > 3") ➞ True

def check(s):

exp=eval(s)

if exp:

return True

else:

return False

print(check("2 < 7 < 15"))

print(check("30 > 45 > 21 > 9"))

print(check("4 < 7 < 8< 12 > 2"))

Question3.

### Examples

replace\_vowels("the aardvark", "#") ➞ "th# ##rdv#rk"

replace\_vowels("minnie mouse", "?") ➞ "m?nn?? m??s?"

replace\_vowels("shakespeare", "\*") ➞ "sh\*k\*sp\*\*r\*"

def replace\_vowels(sent,rep):

vowels="AEIOUaeiou"

for ele in vowels:

sent=sent.replace(ele,rep)

return sent

print(replace\_vowels("the aardvark", "#"))

Question4. Write a function that calculates the **factorial** of a number **recursively**.

### Examples

factorial(5) ➞ 120

factorial(3) ➞ 6

factorial(1) ➞ 1

factorial(0) ➞ 1

def factorial(x):

if x==0 or x==1:

return 1

else:

return x\*factorial(x-1)

print(factorial(3))

**Question 5**

**Hamming distance** is the number of characters that differ between two strings.

To illustrate:

String1: "abcbba"

String2: "abcbda"

Hamming Distance: 1 - "b" vs. "d" is the only difference.

Create a function that computes the **hamming distance** between two strings.

### Examples

hamming\_distance("abcde", "bcdef") ➞ 5

hamming\_distance("abcde", "abcde") ➞ 0

hamming\_distance("strong", "strung") ➞ 1

def hamming\_distance(str1, str2):

i = 0

count = 0

while(i < len(str1)):

if(str1[i] != str2[i]):

count += 1

i += 1

return count

print(hamming\_distance("srina234th","soon456ath"))