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
```

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
```

Question3. Create a function that replaces all the vowels in a string with a specified character.

Examples

```
replace_vowels("the aardvark", "#") ➞ "th# ##rdv#rk"

replace_vowels("minnie mouse", "?") ➞ "m?nn?? m??s?"

replace_vowels("shakespeare", "*") ➞ "sh*k*sp**r*"
```

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
```

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
```