Hamming Distance Problem

Input: Two strings of equal length.

Output: The Hamming distance between these strings.

SAMPLE DATASET:

<u>Input</u>:
```
GGGCCGTTGGT
GGACCGTTGAC
```

<u>Output</u>:
```
3
```

The sample dataset is not actually run on your code.

TEST DATASET 1:

<u>Input</u>:
AAAA
TTTT

<u>Output</u>:
4

This dataset checks if your code isn't keeping count (i.e. returns '0' when the answer is clearly nonzero) or if your code returns a negative value, which is impossible.

TEST DATASET 2:

<u>Input</u>:

```
ACGTACGT
TACGTACG
```

<u>Output</u>:

8

This dataset checks if your code is finding Edit Distance (which would be 2) instead of Hamming Distance.

TEST DATASET 3:

Input:

```
ACGTACGT
CCCCCCCC
```

Output:

```
6
```

This dataset checks if your code is returning the number of matches (2) instead of the number of mismatches (6).

TEST DATASET 4:

Input:
```
ACGTACGT
TGCATGCA
```
Output:
```
8
```

This dataset checks if your code works on a dataset where the two input strings have no matches.

TEST DATASET 5:

<u>Input</u>:

GATAGCAGCTTCTGAACTGGTTACCTGCCGTGAGTAAATTAAAATTTTATTGACTTAGGTCACT
AAATACT

AATAGCAGCTTCTCAACTGGTTACCTCGTATGAGTAAATTAGGTCATTATTGACTCAGGTCACT
AACGTCT

<u>Output</u>:

15

This dataset checks if you have an off-by-one error at the beginning (i.e. you are starting at the second character of the strings instead of the first character).

TEST DATASET 6:

<u>Input</u>:
AGAAACAGACCGCTATGTTCAACGATTTGTTTTATCTCGTCACCGGGATATTGCGGCCACTCAT
CGGTCAGTTGATTACGCAGGGCGTAAATCGCCAGAATCAGGCTG

AGAAACCCACCGCTAAAAACAACGATTTGCGTAGTCAGGTCACCGGGATATTGCGGCCACTAAG
GCCTTGGATGATTACGCAGAACGTATTGACCCAGAATCAGGCTC

<u>Output</u>:
28

       This dataset checks if you have an off-by-one error at the end (i.e. you are ending at the second-to-last character of the strings instead of the last character).