

## Edit Distance

3/4 points (75%)

Quiz, 4 questions

 **Congratulations! You passed!**[Next Item](#)1 / 1  
points

1.

How many insertions are needed to make **axybc** from **abc**?

2

**Correct**Insert **x** between **a** and **b**, then **y** between **x** and **b**.

1



3

1 / 1  
points

2.

What is the edit distance between words **bread** and **really**?

6



3



4

**Correct**Delete **b**, then change **d** to **l**, then insert **l** and **y** in the end.1 / 1  
points

3.

What is the edit distance between **bread** and **really** if it is allowed to insert and delete symbols, but forbidden to replace symbols?

5



**Correct**Remove **b**, remove **d**, insert **l**, **l** and **y**.

## Edit Distance

**3/4 points (75%)**

Quiz, 4 questions

☐ 6☐ 40 / 1  
points

4.

*(This is an advanced problem)*

We want to compute not only the edit distance  $d$  between two words, but also the number of ways to edit the first word to get the second word using the minimum number  $d$  of edits. Two ways are considered different if there is such  $i, 1 \leq i \leq d$  that on the  $i$ -th step the edits in these ways are different.

To solve this problem, in addition to computing array  $T$  with edit distances between prefixes of the first and second word, we compute array  $ways$ , such that  $ways[i, j]$  = *the number of ways to edit the prefix of length  $i$  of the first word to get the prefix of length  $j$  of the second word using the minimum possible number of edits.*

Which is the correct way to compute  $ways[i, j]$  based on the previously computed values?



```

1 ways[i, j] = 0
2 if T[i, j] == T[i - 1, j] + 1:
3     ways[i, j] += ways[i - 1, j]
4 if T[i, j] == T[i, j - 1] + 1:
5     ways[i, j] += ways[i, j - 1]
6 if word1[i] == word2[j] and T[i, j] == T[i - 1, j - 1]:
7     ways[i, j] += ways[i - 1, j - 1]
8 if T[i, j] == T[i - 1, j - 1] + 1:
9     ways[i, j] += ways[i - 1, j - 1]
```



```

1 ways[i, j] = 0
2 ways[i, j] += ways[i - 1, j]
3 ways[i, j] += ways[i, j - 1]
4 ways[i, j] += ways[i - 1, j - 1]
5 ways[i, j] += ways[i - 1, j - 1]
```



```

1 ways[i, j] = 0
2 if T[i, j] == T[i - 1, j] + 1:
3     ways[i, j] += ways[i - 1, j]
4 if T[i, j] == T[i, j - 1] + 1:
5     ways[i, j] += ways[i, j - 1]
```



```

1 ways[i, j] = 0
2 if T[i, j] == T[i - 1, j] + 1:
3     ways[i, j] += ways[i - 1, j]
4 if T[i, j] == T[i, j - 1] + 1:
5     ways[i, j] += ways[i, j - 1]
6 if word1[i] == word2[j] and T[i, j] == T[i - 1, j - 1]:
7     ways[i, j] += ways[i - 1, j - 1]
```

**This should not be selected**

This code doesn't account for ways that replace the last symbol of the  $i$ -th prefix of the first word with the last symbol of the  $j$ -th prefix of the second word.