

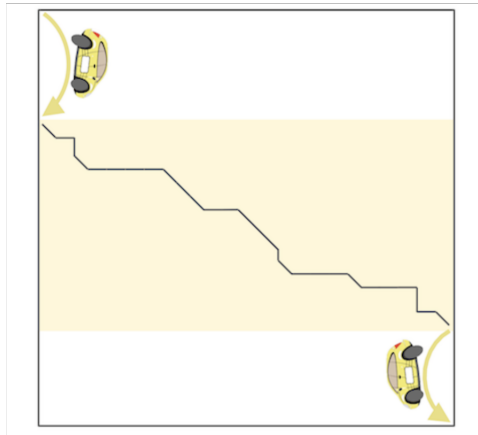
5H Find a Highest-Scoring Fitting Alignment of Two Strings

Fitting Alignment Problem

Find a highest-scoring fitting alignment between two strings.

Input: Two amino acid strings.

Output: A highest-scoring fitting alignment of these two strings (with respect to the scoring parameters) and its score.



Formatting

Input: Two space-separated DNA strings v and w .

Output: The maximum local alignment score of v and w as an integer followed by a newline-separated local alignment of v and w achieving this maximum score (if multiple fitting alignments achieving the maximum score exist, you may return any one). Use the BLOSUM62 scoring matrix and an indel penalty of 1.

Constraints

- The lengths of v will be between 1 and 10^4
- The length of w will be between 1 and 10^3 .
- Both v and w will be DNA strings.

Test Cases

Case 1

Description: The sample dataset is not actually run on your code.

Input:

DISCREPANTLY
PATENT

Output:

2
PA--NT
PATENT

Case 2

Description: This test makes sure that your dynamic programming matrix is correctly initialized. There should be no score punishment for starting at an arbitrary position in string v . Additionally, indels outside of string w should not be reported in the final alignment. If your reconstructed alignment is the whole alignment below instead of the darkened portions then double check that your alignment reconstruction implementation does not include characters that do not fall within string w . If your code outputs a score of 0 then make sure that the base cases in your dynamic programming matrix are correctly set.

Input:

ARKANSAS
SASS

Output:

2
SAS-
SASS

Case 3

Description: This test makes sure that your dynamic programming matrix is correctly initialized. There should be no score punishment for stopping at an arbitrary position in string v . Additionally, indels outside of string w should not be reported in the final alignment. If your reconstructed alignment is the whole alignment below instead of the darkened portions then double check that your alignment reconstruction implementation does not include characters that do not fall within string w . If your code outputs a score of 0 then make sure that the base cases in your dynamic programming matrix are correctly set.

Input:

```
DISCREPANTLY
DISCRETE
```

Output:

```
4
DISCRE--
DISCRETE
```

Case 4

Description: This test makes sure that your code chooses the correct cell in the dynamic programming matrix as the final score for the fitting alignment. If your implementation outputs the score in the bottom right corner of the matrix as it would be done in global alignment you will get a score of 0. Also be sure that your code correctly backtracks from the final score cell and reconstructs the alignment.

Input:

```
CANT
CA
```

Output:

```
2
CA
CA
```

Case 5

Description: A larger dataset of the same size as that provided by the randomized autograder. Check input/output folders for this dataset.