ALIGNMENT(x, y, score)

Input: x is a known string, y is an unknown string, score is a matrix used as a scoring function for alignment

output: The maximum score of aligning x and y

Definition: $T(i,j)$ is the maximum score of aligning $(x_0 \dots x_i)$ and $(y_0 \dots y_j)$

Base Case: (compare to ϕ)

$T(i,-1) + score[x_i, "-"]$ from 0 to i where i is len(x)

$T(-1,j) + score["-", y_j]$ from 0 to j where j is len(y)

$T(-1,-1) + 0$ b/c ϕ aligns ϕ

Formula:

$$T(i,j) = \max \begin{cases} T(i-1,j-1) + score(x[i], y[j]) & \text{diagonal} \\ T(i-1,j) + score["-", y[j]] & \text{up} \\ T(i,j-1) + score(x[i], "-") & \text{left} \end{cases}$$

soln: $T(i-1, j-1)$ is the max score

Time complexity:

$O(nm)$

where n is the length of first string and m is the length of second string.

Lemma: There exist strings x, y and a scoring function δ such that GREEDY ALIGNMENT produces a lower scoring alignment than optimal

---

Proof by contradiction: There do not exist strings x, y and a scoring function δ such that GREEDY ALIGNMENT produces a lower scoring alignment than optimal.

counter example:

known: sushi

unknown: h sushi

Greedy score: δ => sushi - hsushi

DP score: 21 => - sushi
h sushi

By contradiction, lemma is true.

Lemma: There exist strings x, y and a scoring function δ such that GreedyAlignment aligns every character with a mismatch character, even though matching characters exist (and would score higher).

Proof by example

---

hguifar

guifar

Greedy:
h g ui far
g u i f ar -

DP:
h g ui far
- g ui far

By example, lemma is true