Recursive Solution

Turn the 3 possibilities into 3 cases of a recurrence:

$$OPT(i,j) = \min \begin{cases} \text{cost}(a_i, b_j) + OPT(i-1, j-1) & \text{match } a_i, b_j \\ \text{gap} + OPT(i-1, j) & a_i \text{ is not matched} \\ \text{gap} + OPT(i, j-1) & b_j \text{ is not matched} \end{cases}$$

↑ Cost of the optimal alignment between $a_1 \ldots a_i$ and $b_1 \ldots b_j$

↑ Written in terms of the costs of smaller problems

Key: we don’t know which of the 3 possibilities is the right one, so we try them all.

Base case: $OPT(i,0) = i \times \text{gap}$ and $OPT(0,j) = j \times \text{gap}$

(Aligning *i* characters to 0 characters must use *i* gaps.)