Problem 1

(a) \((12/16 \times -1) + (4/16 \times 1) = -1/2\). The matrix is valid for local alignment because the expected score is negative, and there is at least one positive score.

(b) \(3/4 \exp(-\lambda) + 1/4 \exp(\lambda) = 1\). Let \(x = \exp(\lambda)\). We have
\[
\begin{align*}
3/(4x) + x/4 &= 1 \\
x^2 - 4x + 3 &= 0 \\
(x - 3)(x - 1) &= 0
\end{align*}
\]

so \(x = 3\) or \(x = 1\). and \(\lambda = \ln 3\) or \(\lambda = \ln 1 = 0\). \(\lambda\) is positive solution, so \(\lambda = \ln 3 = 1.099\).

(c) Background frequency for AA: 1/16.

Target frequency: \(1/16 \times \exp(\lambda \times 1) = 1/16 \exp(\ln 3) = 3/16\).

(d) Matching pairs: Background freq. 1/4 Target freq. 3/4.

Mismatching pairs: Background freq. 3/4 Target freq. 1/4.

(e) 0.75 \times \log_2(0.75/0.25) + 0.25 \log_2(0.25/0.75) = 0.5 \times \log_2(3) = 0.7925\) bits.

Problem 2

If \(S\) is the normalized bit score, \(E = N/2^S = 10^3 \times 10^9 / 2^{35} = 29.1\). Alternatively, \(E = 10^{12} / 2^{35} \approx 2^{40} / 2^{35} \approx 2^5 = 32\).

With 49 bits: \(E = 10^{12} / 2^{49} = 0.00178\). Alternatively, \(E \approx 2^{-9} / 500 = 0.002\).

Problem 3

Omitting base and edge cases, the main recurrences are below. \(A[i,j]\) is the best score for the substring \(i\ldots j\) assuming that \(i - 1, j + 1\) are not paired. \(P[i,j]\) is the best score assuming that they are paired.

\[
A[i,j] = \max_i \left\{ A[i,j-1] \middle| A[i,t-1] + P[t+1,j-1] + g(x_i,x_j) \right\}
\]

\[
P[i,j] = \max_i \left\{ A[i,j-1] \middle| A[i,t-1] + P[t+1,j-1] + f(x_i,x_j | x_{t-1},x_{j+1}) \right\}
\]

Problem 4

Build a generalized suffix tree for \(S_1,\ldots,S_K\). Let \(w(u) = i\) if node \(u\) represents a suffix of \(S_i\). Let \(C(v) = \vert \{w(u) : u \text{ is a descendent of } v\} \vert\) (that is the number of different strings represented under \(v\)). \(\ell(k)\) is the length of the deepest node with \(C(v) \geq k\). \(C(v)\) can be computed by keeping a \(K\)-long bit vector \(b(v)\) at each node \(v\). When \(v\) is a leaf, \(b(v)\) has a “1” at position \(w(v)\). For a non-leaf \(v\), \(b(v)\) is the AND of the vectors at its children.

Problem 5

Every leaf is not left diverse because it represents a suffix of the string. Label each leaf \(u\) with the unique character \(L(u)\) that occurs before \(\text{str}(u)\). Recursively define \(L(v)\) for internal nodes \(v\) as:

\[
L(v) = \begin{cases} 
  a & \text{if } L(u) = a \text{ for all children } u \text{ of } v \\
  \$ & \text{otherwise}
\end{cases}
\]

Case 1 above is when the left character preceding all the strings under \(v\) is the same (and therefore \(v\) is not left diverse). Nodes with \(L(v) = \$\) are left diverse.