1. **Scoring matrices.** Assume the background frequencies for all four nucleotides are equal, and consider the DNA substitution matrix which gives all matches a score +1 and all mismatches a score $-1$.

   (a) What is the expected score for this matrix? Is this a valid matrix for local alignment? Why?
   
   (b) What is $\lambda$ for this matrix?
   
   (c) What are the background and target frequencies for the aligned pair of nucleotides AA?
   
   (d) Combined into two sets, what are the background and target frequencies for all matching and all mismatching pairs of nucleotides.
   
   (e) What is the relative entropy for this matrix, expressed in bits?

2. **Alignment scores.** Using a query protein sequence of length 1000, you perform a BLAST search of a protein database, containing sequences with one billion ($10^9$) total residues (amino acids). Approximately how many local alignments do you expect to find with normalized score $\geq 35$ bits? What is the approximate E-value of an alignment with normalized score 49 bits?

3. **RNA folding.** Suppose you are given a function $f(x_i, x_j | x_{i+1}, y_{j-1})$ where $x_i, x_{i+1}, y_{j-1}, y_j \in \{A, C, G, T\}$ that equals the energy of simultaneously pairing $x_i$ with $y_j$ given that $x_{i+1}$ is paired with $y_{j-1}$. That is, instead of a simple pairwise energy, the benefit of a pairing depends on the adjacent pair. You are also given a function $g(x_i, y_j)$ that gives the benefit of pairing $x_i$ and $y_j$ assuming that $x_{i+1}, y_{j-1}$ are not paired. (The $i$s and $j$s are just names — the values of the function do not depend on the location in the sequence, only the identities of the nucleotides.)

   Give a dynamic programming solution to find the lowest-cost set of pairings, according to $f$ and $g$ that obey the RNA folding rules.

4. **Suffix trees.** You are given $K$ strings $S_1, \ldots, S_K$ of total length $n$. Define $\ell(k)$ for $k = 2, \ldots, K$ to be the length of the longest substring that is common to at least $k$ of the strings $S_1, \ldots, S_K$. For example, if the strings are *ration*, *natural*, *nation*, then $\ell(3) = 2$ (at), $\ell(2) = 5$ (ation).

   Give an $O(Kn)$-time algorithm to compute $\ell(k)$ for all $k = 2, \ldots, K$.

5. **Suffix trees.** Let $T$ be a suffix tree for string $S$. Let $\text{str}(u)$ be the string that is spelled out when walking from the root of $T$ to a node $u$. A node in $T$ is called left diverse if the occurrences of $\text{str}(u)$ in $S$ are not always preceded by the same character. For example, if $S = ababa\bar{c}$ then the node representing $ba$ is not left diverse since $ba$ is always preceded by $a$. But the node with $\text{str}(u) = b$ is left diverse because sometimes $b$ is preceded by $a$ and sometimes by $c$.

   Give an $O(|S|)$ algorithm to identify all the left-diverse nodes in $T$. 