1. Show that edit distance is a metric, i.e., that it satisfies the metric axioms:
   - \( ed(A, B) \geq 0 \)
   - \( ed(A, B) = 0 \) if and only if \( A = B \)
   - \( ed(A, B) = ed(B, A) \) (symmetry)
   - \( ed(A, C) \leq ed(A, B) + ed(B, C) \) (triangle inequality)

2. Let \( \Sigma = \{a, b, c\} \). Define the function \( \gamma : \Sigma \times \Sigma \rightarrow \mathbb{R}_{\geq 0} \) as follows
   \[
   \begin{align*}
   \gamma(a, a) &= \gamma(b, b) = \gamma(c, c) = 0 \\
   \gamma(a, b) &= \gamma(b, c) = \gamma(c, a) = 0.5 \\
   \gamma(b, a) &= \gamma(c, b) = \gamma(a, c) = 1.5
   \end{align*}
   \]
   Let \( ed_\gamma \) be a weighted edit distance, where the cost of substituting a character \( x \) with a character \( y \) is \( \gamma(x, y) \). The cost of insertions and deletions is 1.
   
   (a) It might seem that we can compute \( ed_\gamma(A, B) \) using the recurrence for the standard edit distance (page 112 on the lecture notes) except \( \delta \) is replaced by \( \gamma \). Show that this is not the case by providing an example for which the recurrence produces an incorrect distance.
   
   (b) Is \( ed_\gamma \) a metric?

3. Describe a family of string pairs \( (A_i, B_i) \), \( i \in \mathbb{N} \), such that \( |A_i| = |B_i| \geq i \) and there is at least \( i \) different optimal edit sequences corresponding to \( ed(A_i, B_i) \). Can you find a family, where the number of edit sequences grows much faster than the lengths of the strings?

4. A string \( S \) is a subsequence of a string \( T \) if we can construct \( S \) by deleting characters from \( T \). Let \( lcss(A, B) \) denote the length of the longest common subsequence of the strings \( A \) and \( B \). For example, \( lcss(\text{berlin, helsinki}) = 4 \) since \( \text{elin} \) is a subsequence of both strings.
   
   (a) Let \( ed_{\text{indel}}(A, B) \) be a variant of the edit distance, where insertions and deletions (indels) are the only edit operations allowed (i.e., no substitutions). Show that
   \[ ed_{\text{indel}}(A, B) = |A| + |B| - 2 \cdot lcss(A, B) \]
   
   (b) Give an algorithm for computing \( lcss(A, B) \) in time \( O(|A||B|) \).

5. Give a proof for Lemma 3.15 in the lecture notes.

6. Let \( P = \text{evete} \) and \( T = \text{neeteneeveteen} \).
   
   (a) Use Ukkonen’s cut-off algorithm to find the occurrences of \( P \) in \( T \) for \( k = 1 \).
   
   (b) Simulate the operation of Myers’ bitparallel algorithm when it computes column 5 for \( P \) and \( T \).