Universität Bielefeld
Technische Fakultät

AG Genominformatik
Prof. Dr. Jens Stoye
Dr. Lucas Robidou

Algorithms in Genome Research

Winter 2024/2025

Exercises

Number 9, Discussion: 2025 January 10

1. Given a list of peaks from a Tandem Mass Spectrum (MS/MS) for peptide de-novo sequencing, one important obstacle for recovering the peptide sequence is to assign peaks to the main ion types b and y (prefix and suffix strings of the peptide sequence, respectively).

   If we know that only b-ions are present in the spectrum, then recovering the sequence becomes simple. Describe an algorithm to do so: Input are the parent mass W and an ordered list of prefix masses $w_1, \ldots, w_k$. (Some peaks may be missing, though.)

2. We modify the above problem such that there are "noise peaks" (of unknown origin) in the mass spectrum. Describe an algorithm that finds a peptide sequence maximizing the number of explained peaks. The algorithm should run in $O(k|\Sigma|)$ time where k is the number of peaks and Σ is the underlying alphabet of amino acids. Hint: Use dynamic programming.

3. Word puzzling:

   - How many different words can be built by using all the letters of the string $S = \texttt{GLÜHWEIN}$ exactly once? Compute the actual value.
   - How many different words can be built by using all the letters of the string $S = \texttt{TEELICHT}$ exactly once? Note that all words need to have the same length and must use the letters the specified number of times: For `ABA`, there are three such words, `AAB`, `ABA`, `BAA`.
   - Try and find a general formula for the number of different words wordnum(S) that can be created from the letters of a string S. Hint: For $S = \texttt{GLÜHWEIN}$, the formula depends only on the length of S, but not for $S = \texttt{TEELICHT}$ - what else does it depend on?

4. Suppose that we do not know the order of characters in a string: For example, the strings `AACCC`, `ACACC`, ..., `CCCAA` are indistinguishable to us. We call such "strings without order" *compomers* (denoted $\texttt{A}_2\texttt{C}_3$ for our example). The *length* of a compomer is the length of the corresponding string (5 in our example).

   - Let $\Sigma = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}\}$ be our alphabet, then there exist 4 compomers of length 1 ($\texttt{A}_1, \texttt{C}_1, \texttt{G}_1, \texttt{T}_1$) and 10 compomers of length 2 ($\texttt{A}_2, \texttt{A}_1\texttt{C}_1, \texttt{A}_1\texttt{G}_1, \texttt{A}_1\texttt{T}_1, \texttt{C}_2, \texttt{C}_1\texttt{G}_1, \texttt{C}_1\texttt{T}_1, \texttt{G}_2, \texttt{G}_1\texttt{T}_1, \texttt{T}_2$). How many compomers exist of lengths 3 and 4?
   - Derive a general formula for the number of compomers of length n over an arbitrary alphabet Σ of size σ.