**Multiple Sequence Alignment**

**Input:**  *k* protein amino acid sequences (*k*= 3 … 5)

**Goal:** Perform a multiple sequence alignment for the given *k* protein sequences.

**Outputs:**

**1**. Visualization of guide tree (output of Step2)

**2.** Multiple sequence alignment of *k* sequences (output of Step3)

**Heuristic:** Progressive alignment (similar to Clustal).

There are 3 main steps of a progressive alignment method.

**Step 1**

- Find pairwise sequence alignments for all possible pairs of sequences.

- Implement a global alignment to find each pairwise sequence alignment (i.e., fill the dynamic programming table).

- Gap penalty will be asked to the user – input parameter.

- Match and mismatch scores for amino acids will be taken from the BLOSUM62 matrix.

- Compute similarity score for each pairwise alignment. A higher score represents the more similar sequences.

*Similarity Score (si , sj ) = # of exact matches / aligned sequence length*

- Construct similarity matrix (*kxk*) by using similarity scores.

**Step 2**

- Build a guide tree by using similarity matrix, which was calculated in the Step 1.

- Use *UPGMA* method while creating a tree branch i.e., group together the most similar sequences at each iteration.

**Step 3**

- Merge alignments by starting from the most similar ones to obtain the final multiple alignment of *k* sequences.

- Use the guide tree to decide the order of merge operations.

- You will compute the edit table and use the backtrack matrix to align each new sequence to the current multiple alignment.

- If there is a gap in an alignment, keep and reflect it to the next alignments, so apply “once a gap, always a gap” strategy.