

Choose any *three* problems. Please typeset your solutions.

Problem 1

We are interested in the edit distance problem (i.e. unit-cost substitution/insertion/deletion). Recall the simple dynamic programming formulation. Read the Gusfield's notes on the edit distance problem.

In the Four-Russians class notes, it shows that the values in the DP table $D[i, j]$ along a line (horizontal, vertical or diagonal in the increasing direction) can increase or decrease by at most *one*.

Now prove something stronger: $D[i, j]$ can **not** decrease along diagonals. That is, $D[i, j] \leq D[i + 1, j + 1]$ for all i, j (assuming within range).

Problem 2

Read the notes on Four-Russians method carefully. In class, when we cover Four-Russians algorithm, a question was raised about how to find the optimal edit script (instead just a score). Tell me how to find optimal edit script within the framework we discussed in class, and state what the time and space needed to find the optimal edit script.

Problem 3

Read carefully (up to page 12) the paper by Keich, et al. on the algorithm that computes the probability of a seed hitting a region (pages 9-10). Now, tell me the difference between Keich et al.'s and what was discussed in class (also in the notes). Briefly explain (in your words) equations (7,8,9) in Keich, et al. (again, on pages 9-10). Also briefly state how you think the algorithm covered in class can be improved in terms of efficiency by using ideas from Keich, et al.

Problem 4

This is essentially the Exercise 14 on p.367 in Gusfield's book.

We went over the Gusfield's MSA approximation in class. The method has polynomial-time running time. When the number of sequences and length of sequences grow, the method may be slow because it computes pairwise alignment between *all* pairs of sequences. A little surprising fact is that we do not actually need to find the *best* center sequence to get a reasonable performance. Suppose we *randomly* pick a sequence as center, and consider the alignment score we get. **Note:** alignment score is what we have on the star tree, *not* the sum of pair scores. That is, score is $\sum_j D[i, j]$.

1. Show that the average value of alignment score restricted to a star tree is no more than $2M$, where $M = \min_i (\sum_j D[i, j])$ over all i , is alignment score of the best center.

2. Show that median of alignment score on a star tree is at most $3M$, and argue that we expect to pick only two center strings to get an alignment whose score on a star tree is no more than $3M$.