Assignment 6. Miller-Myers in linear space

Miller-Myers in linear space

If possible, develop a linear-space modification of the Miller-Myers algorithm for computing edit distance. The original paper “A file comparison problem” is here. It includes a pseudocode. This algorithm was also discussed in class (Lecture 5.4). Here is an improved version of the algorithm with a cleaner pseudocode: link. Just implementing this algorithm in C and testing its correctness with a stress test will give you 50%.

You might think of how to reduce the space complexity in the spirit of Hirshberg's ‘divide and conquer’ approach (see Lecture 5.3), but the solution is not straightforward and requires thinking outside the box.

An implementation and correctness testing might give you an additional 100% bonus, but it is not required.

If you think that it is impossible to perform the Miller-Myers algorithm using only a linear space, give detailed explanation and justification of why this is impossible.

Customizable Hirshberg’s algorithm

(alternative version of the assignment)

If you find the previous assignment too challenging, then you may alternatively implement an original Hirshberg’s algorithm which computes an edit distance in linear space (for the 40% of the grade). Then you should modify the implementation in such a way that it accepts two strings and a scoring matrix and is able to compute any variation of alignment according to this matrix, either by finding a cheapest path or a greediest path in the edit graph (for the remaining 60%).