The O(mn)-time algorithm for the longest-common-subsequence problem appears to be a folk algorithm. Knuth [1] posed the question of whether subquadratic algorithms for the LCS problem exist. Masek and Paterson [2] answered this question in the affirmative by giving an algorithm that runs in O(mn/\lg n) time, where n≤m and the sequences are drawn from a set of bounded size. For the special case in which no element appears more than once in an input sequence, Szymanski [3] shows how to solve the problem in O((n+m)(\lg(n+m)) time. Many of these results extend to the problem of computing string edit distances (Problem 15-5).

If you want to learn more about this image-compression technique (Problem 15-8), please refer to [4]. A quick introductive video can be found at: https://www.youtube.com/watch?v=vIFCV2spKtg


Rules for ALL HWs (in addition to the statements in the syllabus):
You are encouraged to discuss the problem sets and study together in group, but when it comes to formulating/writing solutions you must work alone independently; i.e., you should be able to explain your answer clearly to anyone else. Note that this says discuss in group — copying homework solutions from another student, from the Internet, solution sets of friends who have taken this course or one similar to it previously, or other sources will be considered cheating and referred to the student attorney general. You must include a signed honor statement with each submission explicitly listing the people you worked with and stating that you completed the assignment in accordance with these rules.