

Please feel free to discuss these problems on the unit discussion board or directly with your colleagues. If you would like to have your answers marked, please either hand them in in person at a lecture or problems class. Submitted work will be marked as quickly as possible, ideally within one week of being handed in.

1. (From Gusfield) Given a set S of k strings, we want to find every string in S that is a substring of some other string in S . That is, we want to find the smallest subset of strings $S' \subset S$ such that no string in $S - S'$ is a substring of any other string in that set. Assuming that the total length of all the strings is n , give an $O(n)$ -time algorithm to solve this problem.
2. (From Gusfield) A suffix tree for a string S can be viewed as a keyword tree, where the strings in the keyword tree are the suffixes of S . In this way, a suffix tree is useful in efficiently building a keyword tree when the strings for the tree are only *implicitly* specified. Now consider the following implicitly specified set of strings : Given two strings S_1 and S_2 , let D be the set of all substrings of S_1 that are not contained in S_2 . Assuming the two strings are of length n , show how to construct a keyword tree for set D in $O(n)$ time.
3. Assume you are given two strings S_1 and S_2 of length n . Describe how you can find the length of the longest substring in S_2 that is also a substring of S_1 in $O(n)$ time. You should describe the different parts of your algorithmic solution in detail and also give the total running time.
4. This question is about the LCA problem. For the example tree in the lectures with 11 nodes, write out the complete reduction to the ± 1 RMQ problem. What is the total size of the data created by the LCA preprocessing?
5. If you are given an independent and efficient solution to the full RMQ problem, how can you reduce the space needed for the arrays created in the reduction from LCA to RMQ? Show your new reduction for the example tree in the lectures with 11 nodes.