
Homework 7 – Due Wednesday, October 24, 2007

Please refer to the general information handout for the full homework policy and options.

Reminders

- Your solutions are due *before* the lecture. Late homework will not be accepted.
- Collaboration is permitted, but you must write the solutions *by yourself without assistance*, and be ready to explain them orally to a member of the course staff if asked. You must also identify your collaborators. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.
- To facilitate grading, please write down your solution to each problem on a separate sheet of paper. Make sure to include all identifying information and your collaborators on each sheet. Your solutions to different problems will be graded separately, possibly by different people, and returned to you independently of each other.
- *For all problems where you are asked to design an algorithm, do not forget to prove correctness and analyze your algorithm's time and space complexity.*

Exercises These should not be handed in, but the material they cover may appear on exams:

1. Determine an LCS of 1,0,0,1,0,1,0,1 and 0,1,0,1,1,0,1,1,0
2. Write pseudocode for the procedure that can reconstruct an LCS from the table of optimal lengths, discussed in class, in time $O(m + n)$.
3. Give a memoized version of the LCS algorithm discussed in class.
4. Explain how to modify the LCS algorithm discussed in class, so that it computes the length of a LCS (but not the subsequence itself) using $O(\min(m, n))$ space. The running time should still be $O(mn)$.
5. Write pseudocode for the procedure that can reconstruct an optimal RNA secondary structure from the dynamic programming table of optimal lengths, discussed in class. What is the running time of your procedure? What additional information do you need to store in the table to speed this procedure up?
6. Chapter 6, problem 1.

Problems to be handed in

1. (**Order of phone calls**) Chapter 6, problem 16. *Hint:* Instead of building a table as we have done in all our dynamic programming examples, think of labeling the nodes of the tree. In which order will you consider the nodes? How do you compute the label(s) for each node from previously computed labels?

2. (**Longest rising trend**) Chapter 6, problem 17.
3. (**Time management**) Chapter 6, problem 20. *Hint:* First compute h_{ij} , the minimum number of hours needed to earn grade j in course i , for all i from 1 to n and all j from 1 to g .