MAT-72006 Advanced Algorithms and Data Structures
October 13, 2016
HW 6: 15 Dynamic Programming

1.

Explain shortly the difference between dynamic programming (DP) and divide-and-conquer (D&C). For concreteness, consider e.g., mergesort. Why doesn't mergesort follow the DP paradigm (but is a D&C algorithm instead)?

2.

What is the main difference between a dynamic programming algorithm and a greedy algorithm? In particular, explain what "optimal substructure" means (if anything) for either paradigm.

3.

Recall the rod cutting problem. Give a counterexample for showing the following greedy strategy does not always determine an optimal way to cut rods. Define the *density* of a rod of length i to be p_i/i, that is, its value per inch. The greedy strategy for a rod of length n cuts off a first piece of length i, where $1 \leq i \leq n$, having a maximum density. It then continues by applying the greedy strategy to the remaining piece of length $n - i$.

4.

Consider a modification of the rod-cutting problem in which, in addition to a price p_i for each rod, each cut incurs a fixed cost of c. The revenue associated with a solution is now the sum of the prices of the pieces minus the costs of making the cuts. Give a dynamic-programming algorithm to solve this modified problem.

5.

The Fibonacci numbers are defined by the recurrence $F_0 = 0$, $F_1 = 1$, and $F_i = F_{i-1} + F_{i-2}$ for $i \geq 2$. Give a linear-time dynamic-programming algorithm for computing the nth Fibonacci number.