

# Study Guide for Dynamic Programming

## CISC 365, Fall 2010

You should be able to do the following

1. Write a recursive solution to a stated problem; or read and understand a recursive solution that is given to you. Study the following examples.
  - a. Rod cutting: equation given on page 362 (equation 15.2). The corresponding recursive code is on page 363. (If you were given the equation on page 362, you should be able to write recursive code from that.)
  - b. Matrix-chain multiply: equation given on page 374 (equation 15.7). The corresponding recursive code is shown on page 385. (If you were given the equation on page 374, you should be able to write recursive code from that.)
  - c. Longest common subsequence: equation given on page 393. (Recursive code is not shown in the text. Practice writing this.)
2. Understand how “memoization” can be used to make the recursive code run in polynomial time rather than exponential time. You should be able to write code like this: given recursive code, add the memoization. Study the examples: rod cutting (memoized code on pages 365-366), matrix-chain multiply (memoized code on page 388). The textbook does not show memoized code for longest common subsequence; practice writing this. Note that for matrix chain multiply, the recursive code on page 385 has exponential runtime, and the memoized code on page 388 has  $O(n^3)$  runtime; be able to explain why this is.
3. Fill in the cost table for the matrix-chain multiply example. Refer to Prof. Blostein’s handout on matrix-chain multiply and CYK parsing (posted on the website). Filling in a whole table takes a long time, but you should be prepared to fill in a missing entry; practice on the three missing entries on page 2 of Prof. Blostein’s handout.

I will not expect you to do the following on a test:

1. Write or read code for a non-recursive dynamic programming algorithm, such as the one for matrix-chain multiply (page 375) or longest-common subsequence (page 394).
2. Answer questions on CYK parsing. This is a supplemental example, use it to deepen your understanding of the general algorithm-framework used by matrix-chain multiply.