DAT600: Algorithm Theory

Assignment - 4: Dynamic Programming

Submission Deadline:	

Problem-1: Matrix Multiplication

The table given below shows the dimensions of five matrices:

Matrix	*Dimension*
A_1	30 x 35
A_2	35 x 15
A_3	15 x 5
A_4	5 x 10
A_5	10 x 25

This problem is to find **the most <u>expensive</u> way** of multiplying the matrices A_1 to A_5.

- Parenthesize the product A_{15} such that the total number of scalar multiplications is **maximized**
- Verify whether your answer for a) is correct by modifying the MATLAB code for "matrix chain multiplication".

Problem-2: Extended Fibonacci Series (Exam Q5, May 2014)

A series of number ***F(n)*** is generated by the following formula:

n=0: F(0) = 0
n=1: F(1) = 1
n=2: F(2) = 2
n>2: F(n) = (F(n-1) * F(n-2)) + (n-3) * F(n-3)

- Propose a divide-and-conquer (DaQ) based recursive algorithm to find *F(n)*.
- Show the "subproblem graph" for the DaQ solution, and determine the time taken in terms of big-O notation.
- Propose a dynamic programming based solution.
- Show the "subproblem graph" for the dynamic programming based solution; determine the time taken in terms of big-O notation.

Problem-3: LCS (Exam Q2, Feb 2012)

This problem is about finding the Largest-Common-Subsequence (LCS) between two sequences of alphabets (you may take subsequences as text substrings).

- For finding LCS, explain why dynamic programming is efficient.
- Explain the steps involved in developing an algorithm based on dynamic programming.
- Using the sequences "CACAQ" and "CADACA" show how the dynamic programming based algorithm for LCS works (it is suffice to show the tables).