Homework 4: Dynamic Programming
Due Date: November 16, 2017

You are allowed to do this homework individually, or in groups of two. Each group will return a single homework and will get the same grade. But a group is NOT allowed to share anything written with another group. Remember that each group member has to fully participate in the solution and understand the solution to perform well in an exam.

Problem 1. Given a set $S$ of $n$ integers and an integer $T$, give an algorithm to check whether there exists a subset $A$ of $S$ such that the sum of the integers in $A$ is equal to $T$. Give the DP Formulation, clearly explain how you derived it, and give pseudocode. Run your algorithm on a small example by constructing the memoization table.

Problem 2. Given a wooden block of length $n$ meters, we would like to cut it into multiple pieces and sell each piece. A $i$ meter long piece worths $d_i$ dollars. Write a dynamic programming algorithm to determine the maximum value we can make by cutting up the wooden block and selling each piece. For example, if we have a block of length 4 and $d_2 = 5$, $d_3 = 8$, $d_4 = 9$, the maximum value we can obtain is 10, by cutting the block into two pieces of length 2. State the DP Formulation, clearly explain how you derived it and give pseudocode. Give a small example showing how you constructed the memoization table.

Problem 3. Use the matrix multiplication algorithm to find the optimal way to compute $A_1 A_2 A_3 A_4 A_5$ where $p$=[6,8,4,10,5,3]. Show the table and show how to extract the actual parenthesization that resulted in the optimal cost.

Problem 4. Let $a_1, a_2, \ldots, a_n$ be $n$ distinct keys ordered from smallest to largest. Let $p_i$ denote the probability of searching for key $a_i$. Give a dynamic programming algorithm that constructs a Binary Search Tree $T$ to minimize average search time. The average seach time for $T$ is given by:
\[ \sum_{i=1}^{n} p_i d_i^T \]
where $d_i^T$ denotes the depth of key $a_i$ in $T$. State the DP Formulation, clearly explain how you derived it and give pseudocode. Give a small example showing how you constructed the table.