# CSCI 270 Homework #3

*Due Date: Monday, February 17th, 11:59pm*

1. There are $n$ poor college students who are renting two houses. For every pair of students $p_i$ and $p_j$, the function $d(p_i, p_j)$ outputs an integer between 1 and $n^2$ that indicates the amount of drama that will ensue if both students are placed in the same house. The total drama is $\max_{i,j} d(p_i, p_j)$ over all pairs of students in the same house. That is, drama is not cumulative: it is determined by the worst pair of people.

   Given an integer $k$ as input, design an $O(n^2)$ graph algorithm to determine how you can partition the students such that the total drama $\leq k$, or assert that no solution exists.

2. There are $n$ scaffolds, where the height of the $i$th scaffold is $h_i$. A frog is trying to hop from the 1st to the $n$th scaffold. From the $i$th scaffold, the frog can either jump to the $(i+1)$st scaffold at cost $|h_i - h_{i+1}|$, or to the $(i+2)$nd scaffold at cost $|h_i - h_{i+2}|$.

   Let $FROG(i)$ return the cost of the cheapest path from scaffold $i$ to scaffold $n$. Provide a dynamic programming algorithm to find the cost of the cheapest path for the frog, and analyze the runtime.

3. You are given an input string S[1:n]. You want to find a contiguous substring S[i:j] that forms the longest possible palindrome. If your string is "YABBADABBADOO", the solution would be "ABBADABBA". Design an efficient dynamic programming algorithm to calculate the length of longest possible palindrome, and analyze the runtime.

4. Alice and Bob are playing a game. There is a linked list with $n$ nodes, each with a single integer $l_i$. They take turns choosing either the integer in the head node or tail node (and removing that node from the list) until the list is empty. The final score for a player is the sum of the numbers chosen by that player.

   (a) Give an example where the optimal strategy is **not** to always take the largest number. Explain your answer.

   (b) Design an efficient dynamic programming algorithm to determine the final scores of Alice and Bob. Alice and Bob have a **lot** of problem-solving skills, due to how many CSCI 170 examples they appeared in. Therefore, assume that Alice and Bob both play optimally.

   (c) Analyze the runtime of your algorithm.

   (d) Explain how you can modify your algorithm so that, instead of simply returning the final scores of Alice and Bob, you identify the sequence of moves each player will make.