# CS 5150/6150: Assignment 2
# Due: Sep 12, 2011

This assignment has 6 questions, for a total of 100 points and 0 bonus points. Unless otherwise specified, complete and reasoned arguments will be expected for all answers.

Question 1: Recurrences ..... **[20]**

Solve each of the following recurrences. You may use any method you like, but please show your work. In each recurrence, you may assume convenient starting values for $T(0)$ or $T(1)$ unless otherwise specified.. Note that $c$ is an undetermined constant.

Solving a recurrence means that you provide a bound of the form $T(n) = O(f(n))$ for a specific $f$. Tight bounds get full credit: for example, if the recurrence is $T(n) = 2T(n/2) + cn$, the answer $T(n) = O(n^2)$ is correct but not precise enough, and you will not get full credit for it.

(a) [**4**] $T(n) = 3T(n/2) + cn$

(b) [**4**] $T(n) = 4T(n/4) + n \log n$

(c) [**4**] $T(n) = 5T(n/8) + \sqrt{n}$

(d) [**4**] $T(n) = 2T(n/2) + \frac{n}{\log n}$

(e) [**4**] $T(n) = \frac{T(n-1)}{T(n-2)}$, $T(0) = 1, T(1) = 2$.

Question 2: Finding a local minimum ..... **[10]**

Please solve problem 11 from `http://compgeom.cs.uiuc.edu/~jeffe/teaching/algorithms/notes/01-recursion.pdf`

Question 3: Searching for a target sum ..... **[10]**

Given an array $S$ of $n$ numbers and a query $x$, design an algorithm that determines whether there exist two elements of $S$ that sum to $x$.

Question 4: Hidden Surface Removal ..... **[15]**

You are given $n$ *nonvertical* lines in the plane, with the $i^{\text{th}}$ line specified by the equation $y = a_i x + b_i$. Assume that no three lines meet at a single point. A line $\ell_i$ is *uppermost* at some $x$-coordinate $x_0$ if $a_i x_0 + b_i > a_j x_0 + b_j$ for all $j \neq i$. Intuitively, a line is uppermost at $x_0$ if you can "see" it from above (looking down from $(x_0, \infty)$.

A line is *visible* if there is some $x_0$ for which it is uppermost. Output the set of visible lines as efficiently as possible.

Question 5: Selection ..... **[25]**

Please solve problem 13 (all parts) from `http://compgeom.cs.uiuc.edu/~jeffe/teaching/algorithms/notes/01-recursion.pdf`. The points breakdown is 10, 5, and 10

Question 6: Implementing a high quality selection routine ..... **[20]**

There is often a gap between the theory and practice of algorithm design. We're going to explore that gap with the help of the selection problem. Our goal will be, given input set of numbers $S$ and a parameter $k$, to compute the $k^{\text{th}}$ largest number in $S$. Note that the median corresponds to $k = n/2$.

(a) Implement an algorithm for $k$-selection.

(b) Experiment with different choices for finding a pivot. Specifically, use different groupings of the input and report on how this affects the performance

(c) How does this algorithm compare to sorting and then taking the $k^{\text{th}}$ element in practice ? Provide concrete empirical evidence in support of your answer.

(d) How do random pivots compare to computing deterministic pivots ? Again, experiment with different random pivots and provide concrete empirical evidence in support of your answer.