Summer 2007 **CMSC 351: Homework 2** Clyde Kruskal

Due at the start of class Wednesday, June 13, 2007. There are FOUR problems.

Problem 1. Professor Hilbert invents a program that uses exactly $100n^4$ operations to solve the borogove problem of size n. Professor Poincare invents a program that uses $2^n/10^6$ operations to solve the same problem.

(a) Assume that you are willing to run Hilbert's program for some specific amount of time. Assume that your computer speed increases by a factor of 20. How much larger a problem can you solve in the same amount of time? Show your work.

(b) Assume that your computer executes one billion operations per second. Approximately how large a problem can you solve in a year? Show your work (but not the arithmetic).

(c) Assume that you are willing to run Poincare's program for some specific amount of time. Assume that your computer speed increases by a factor of 20. How much larger a problem can you solve in the same amount of time? Show your work.

(d) Assume that your computer still executes one billion operations per second. Approximately how large a problem can you now solve in a year? Show your work (but not the arithmetic).

(e) Whose program should you use if you are going to run it for a year?

(f) For approximately what values of n is Professor Hilbert's program faster than Professor Poincare's? Show your work.

Problem 2. Show the following. In each of (a), (b), and (c) state specific values of the constants (e.g. c_1, c_2, n_0) you used to satisfy the conditions, and show how you arrived at the values.

(a) $3n^2 + 4n - 5 = O(n^2 - 3n + 1)$

(b) $2n^3 - 5n^2 + 3 = \Theta(n^3)$

(c) $2n^2 + 5n(\lg n)^3 = O(n^2)$ [Hint: Find n_0 such that $(\lg n)^3 \le n$, for all $n \ge n_0$.]

(d) $n \log n = o(n^{1.5})$

Problem 3. For each pair of expressions (A, B) below, indicate whether A is O, o, Ω, ω, or Θ of B. Note that zero, one or more of these relations may hold for a given pair; list all correct ones.

	A	B
(a)	n^{100}	2^n
(b)	$(\log n)^{12}$	$\sqrt{n}$
(c)	$\sqrt{n}$	$n^{\cos(\pi n/8)}$
(d)	10^n	100^n
(e)	$n^{\log n}$	$(\log n)^n$
(f)	$\log(n!)$	$n \log n$

Problem 4. Consider a version of insertion sort where the algorithm skips an element at each iteration of the insertion phase. So when inserting element i, it compares to element $i - 2$, then $i - 4$, then $i - 6$, etc. When element i is finally greater than element $i - 2k$, the algorithm goes back to compare it to element $i - 2k + 1$. Assume there is no sentinel.

Analyze the worst case number of comparisons for this version of insertion sort. Make your answer as simple and elegant as possible. Show your work.