CS/Math 387 Homework for Friday, Week 7

Problem 1. Recall the definition of big-O notation. Suppose $f, g \colon \mathbb{N} \to \mathbb{R}_{\geq 0}$. We say $f(n) = O(g(n))$ if there exists an integer $c \geq 1$ and and a positive integer n_0 such that $f(n) \leq cg(n)$ for all $n \geq n_0$.

(a) Prove from the definition of big-O notation that $n + n^2 = O(n^2)$.

(b) Suppose that $f(n) = O(n) + O(n \log(n))$ (this means that there exist integers $c, c' \geq 1$ and an integer n_0 such that $n \geq n_0$ implies $f(n) \leq cn + c'n \log(n)$). Carefully prove that $f(n) = O(n \log(n))$.

Problem 2. True or False. No proof required.

(a) $2n = O(n)$ (b) $n^2 = O(n)$ (c) $n^2 = O(n \log^2(n))$
(d) $n \log(n) = O(n^2)$ (e) $3^n = 2^{O(n)}$ (f) $2^{2^n} = O(2^{2^n})$.

Problem 3. True or False. No proof required.

(a) $n = o(2n)$ (b) $2n = o(n^2)$ (c) $2^n = o(3^n)$
(d) $1 = o(n)$ (e) $n = o(\log(n))$ (f) $1 = o(1/n)$.

Problem 4.

(a) Show that the complexity class P is closed under the concatenation operation. That is, show that if $A, B \in P$, then $AB \in P$.

(b) Show that the complexity class P is closed under complementation. That is, show that if $A \in P$, then $A := \Sigma^* \setminus A \in P$.

For both proofs, you should provide an outline for an algorithm as in Sipser's text: give a list of stages and show that each step in each stage uses a polynomial number of steps. (Here, as usual, *polynomial number* means polynomial in the input size.) Number your stages, indicate the run times for each step, and indicate the number of iterations of any loop that appears as in Sipser's text using big-O notation. Compute the total run time.

Problem 5. An undirected simple graph G is 3-*colorable* if each of its vertices can be assigned one of three colors in such a way no two adjacent vertices have the same color. (Adjacent vertices are vertices that share an edge.) Define the language

$$\text{3COLOR} = \{\langle G \rangle : G \text{ is 3-colorable}\}.$$

(a) Give some examples of graphs that are 3-colorable and some that are not.

(b) How many ways are there to color a graph with m nodes using 3 colors if there is no restriction on the colors of adjacent vertices?

(c) Show that 3COLOR is in NP by giving a numbered algorithm in the style of those given in Sipser's text. Give the run time for each stage and the total run time in big-O notation.

Problem 6. Let n, b be integers with $n \geq 1$ and $b \geq 2$.

(a) There are two parts to this first problem:

(i) Show that the number of digits of n base b is $\lfloor \log_b(n) \rfloor + 1$.

(ii) Why isn't it the number of digits $\lceil \log_b(n) \rceil$?

(Hint: n has r digits base b if and only if n is between which two powers of b? Be careful with the endpoints of that interval.)

(b) Let $u(n)$ be the length of n written in unary: $n = \overbrace{1 \cdots 1}^{n-\text{times}}$. Show that $u(n) \neq O(\log(n))$.

(c) Consider the operation $d \colon \mathbb{N} \to \mathbb{N}$ defined by $d(n) = \lfloor n/2 \rfloor$. Next, define $h \colon \mathbb{N} \to \mathbb{N}$ by $h(n) = \min\{k : d^{(k)}(n) = 0\}$. Here, $d^{(k)}$ denotes k-fold composition. Is $h(n) = O(\log(n))$? Prove or disprove.

PROBLEM 7. (bonus) Consider the language

$$\text{MODEXP} = \{\langle a, b, c, m \rangle : a, b, c, m \text{ are positive integers written in binary and } a^b = c \bmod m\}.$$

Show that MODEXP $\in P$. Note that the most obvious algorithm does not run in polynomial time. Hint: first try the case where b is a power of 2.