Problem Set 1 – Due Friday, April 3, 2015

Instructions: *Read the course-information sheet.* Remember to acknowledge anyone with whom you discussed problems. Recall too that homeworks are due at 10:45 am in the turn-in box in Kemper 2131, and that late homeworks are not accepted. This problem set has **six** problems (don't miss page 2).

Problem 1 Call a number $x \in \mathbb{N} = \{1, 2, 3, \ldots\}$ a **palindromic number** if, written as a decimal string X without leading zeros, it's a palindrome ($X = X^R$). Write a formula for D_n, the number of n-digit palindromic numbers. By induction, prove your formula correct. What is D_{20}?

Problem 2

(a) Professor Lazybones' computer has two print queues (queue-1 and queue-2) for the same printer. The printer is in a room that's a 1-minute from his office. One or both of the queues goes down unpredictably, and the professor wants to see which of the print queues, if any, is up. The professor has a one-page file that's open on his desktop and decides to use it to test each queue. He could spool the file to queue-1, walk to the printer, see if the printout is there, walk back to his office, and then do it all again, now spooling the file to queue-2. But this will take two trips to the printer room. Describe an alternative strategy that will need only one trip. It should not involve printing anything other than the one open file. Assume the printer does not print header pages and that status of a queue (up or down) doesn't change while the professor is testing it.

(b) Generalize: how do you solve the problem when there are n print queues, $\{1, \ldots, n\}$?

(c) Prove that your (generalized) algorithm is optimal. You will need to define what you mean by optimal, including the class of algorithms with respect to which you claim optimality.

Problem 3. For each of the following counting problems, give not only your answer, but explain how it is computed, justifying any formulas employed. None of these should be overly tedious or require computer calculation.

(a) Let $L = \{\texttt{a}, \texttt{b}, \texttt{ab}\}$. How many strings of length 10 are in L^*?

(b) Let $L = \{\texttt{a}, \texttt{bb}\}$. How many strings of length 10 are in L^*?

Problem 4. Consider the infinite set of numbers $S = \{1, 10, 100, 1000, 10000, \ldots\}$. Prove that there are two numbers in S that differ by a multiple of $N = 314159265359$. *Hint: pigeonhole principle.*

Problem 5 Recall the DIOPHANTINE EQUATION problem: given a multivariate polynomial P with integer coefficients, determine if P has an integer root.

(a) Prof. Rogaway claimed without proof that there is no algorithm to answer this question. But suppose I give you an *oracle* (a "magic box") to answer it. In a single computational step, it says *yes* or *no* according to whether or not P has an integer root. Given such an oracle, describe an algorithm that *finds* an integer root of any multivariate polynomial that has one (and reports *No Root* otherwise).

(b) Let $s(n)$ be the maximum number of computational steps that your algorithm takes to run (on some fixed, oracle-containing computer) when the polynomial P is described by a string of length n. Explain why there is no algorithm to compute $S(n)$ for *any* function S such that $S(n) \geq s(n)$ for all n. (In brief, some functions just grow *too fast* to be computable.)

Problem 6. State whether the following propositions are true or false, carefully explaining each answer.

(a) $\emptyset^*$ is a language.

(b) ε is a language.

(c) Every language is infinite or has an infinite complement.

(d) Some language is infinite and has an infinite complement.

(e) The set of real numbers is a language.

(f) There is a language that is a subset of every language.

(g) The Kleene closure (the star) of a language is always infinite.

(h) The concatenation of an infinite language and a finite language is always infinite.

(i) There is an infinite language L containing the emptystring and such that L^i is a proper subset of L^* for all $i \geq 0$.