Harvard University
Computer Science 121

Problem Set 3

Due Friday, October 17, 2008 at 1:10 PM.
Late problem sets may be turned in until Monday, October 20, 2008 at 1:10 PM with a 20% penalty.
Please hand in Parts A and B separately; each part should be stapled.
All problem sets should be dropped off in the CS 121 box in the basement of Maxwell Dworkin.
See syllabus for collaboration policy.
WRITE YOUR TF's NAME ON EACH PART OF THE PROBLEM SET

PART A (Graded by Prabhas)

PROBLEM 1 (5+5+5+5 points)

Say whether the following languages are (necessarily) regular or (can be) nonregular. Prove your answers.
(A) $L = \{a^{i-j}b | i = 5j\}$
(B) $L = \{w | w \in \{a, b\}^*$ is not a palindrome $\}$. A palindrome is a string that reads the same backwards and forwards, ie. $w = w^R$.
(C) The intersection of finitely many regular languages.
(D) The union of countably many regular languages.

PROBLEM 2 (7+8 points)

(A) Let $\Sigma = \{0, 1, +, =\}$. ADD $= \{x = y+z | x, y, z$ are integers in binary, and x is the sum of y and $z\}$. Prove that ADD is not regular.
(B) Let INTERLEAVE-ADD be the language consisting of string $x_n y_n z_n \ldots x_1 y_1 z_1 x_0 y_0 z_0 \in \{0, 1\}^{3(n+1)}$ such that $x_0 \ldots x_n, y_0 \ldots y_n, z_0 \ldots z_n$ are binary representations of integers x, y, z where $x = y + z$. Prove that INTERLEAVE-ADD is regular.

PROBLEM 3 (Challenge!! 3 points)

Prove that $L = \{w | w$ is a prime number in binary$\}$ is not regular.
(Hint: You might find fermat's little theorem useful. It states that if p is a prime number, then for any integer a such that a, p are relatively prime, $a^{p-1} \bmod p = 1$).

(TURN OVER!)

PART B (Graded by Eleanor)

PROBLEM 4 (6+5+4 points)

Let

$$L_1 = \{a^n b^m a^m b^n : m, n \geq 0\}$$
$$L_2 = \{a^n b^m : m, n \geq 0, m \neq n\}$$

(A) Construct a context-free grammar G_1 for L_1 and a context-free grammar G_2 for L_2.
(B) Construct a context-free grammar for $L_1 L_2$.
(C) Draw a parse tree for the string *aaababbbabb* for the CFG from (B).

PROBLEM 5 (7+7 points)

(A) Consider the alphabet $\Sigma = \{a, b, c, (,), \cup, ^*, \emptyset\}$. Let L be the language of all regular expressions made up of these symbols. Formally, $L = \{\alpha \in \Sigma^* : \alpha$ is a fully parenthesized regular expression over the alphabet $\{a, b, c\}\}$. Show that L is context-free by giving the 4-tuple for the CFG that generates it.
(B) Show that L is *not* regular.

PROBLEM 6 (3+8+5 points)

A context-free grammar G is **ambiguous** if there exists a string $w \in L(G)$ with two distinct leftmost derivations in G.

(A) Show that the context-free grammar $G = (V, \Sigma, R, S)$, where $V = \{S, A, a, b\}$, $\Sigma = \{a, b\}$, and $R = \{S \rightarrow AA, A \rightarrow AAA, A \rightarrow bA, A \rightarrow Ab, A \rightarrow a\}$ is ambiguous, because aba has two different leftmost derivations in G.

(B) A language L is **inherently ambiguous** if all context-free grammars G such that $L = L(G)$ are ambiguous. Show that if L is a regular language, then L is **not** inherently ambiguous. (*Hint*: think about regular grammars).

(C) Show that $L(G)$, where G is the context-free grammar in Part (A), is **not** inherently ambiguous.