Harvard University Extension School
Computer Science E-121

Problem Set 4

Due Friday, October 11, 2013 at 11:59 PM Eastern Time.
Submit your solutions in a single PDF called lastname+ps4.pdf emailed to cscie121@seas.harvard.edu.
LATE PROBLEM SETS WILL NOT BE ACCEPTED.

Problem set by ** ENTER YOUR NAME HERE **

Collaboration Statement: **FILL IN YOUR COLLABORATION STATEMENT HERE (See the syllabus for information)**

See syllabus for collaboration policy.

PROBLEM 1 (2+4 points)

For this problem, you may assume that we are working over the alphabet $\Sigma = \{a, b\}$.
(A) Give a simple English description of the language generated by the following context-free grammar (you do not need to justify your description):

$$S \to \varepsilon \mid aY \mid bX \mid XY$$
$$X \to \varepsilon \mid bX \mid Xa$$
$$Y \to \varepsilon \mid aY \mid Yb.$$

(B) Give a context-free grammar for the following: $L = \{w : w \text{ has twice as many } a\text{'s as } b\text{'s}\}$.

PROBLEM 2 (6 points)

Draw the state diagram for a PDA for the language in Problem 1B. Use the state diagram notation for PDAs given in Sipser.

PROBLEM 3 (8+4 points)

A satisfied boolean expression is a string over the alphabet $\Sigma = \{0, 1, (,), \neg, \wedge, \vee\}$ representing a boolean expression that evaluates to 1, where $\neg$ is the *not* operator, $\wedge$ is the *and* operator, and $\vee$ is the *or* operator (see Sipser p.14). For instance, 1, $(1 \wedge (\neg 0))$, and $(\neg(\neg 1))$ are satisfied boolean expressions.

Meanwhile, $)(1$, $(0\neg)$, $\wedge(0 \vee 1)$, $(0 \wedge 1)$, and $(\neg(1 \vee 0))$ are examples of strings that are *not* satisfied boolean expressions: (The first three examples are strings that aren't valid expressions and simply don't make sense. The last two are well-formed expressions, but they evaluate to 0, rather than to 1.)

(A) Write a context-free grammar that generates the language of satisfied boolean expressions. You may assume that expressions must be completely parenthesized. For instance, your grammar need not generate $\neg(0 \wedge \neg 0 \wedge 1)$ but should generate its equivalent, $(\neg(0 \wedge ((\neg 0) \wedge 1)))$.

(B) Prove that the language of satisfied boolean expressions is not regular.

PROBLEM 4 (8 points)

Let L be the language $\{w : w \text{ has equal numbers of } a\text{'s}, b\text{'s and } c\text{'s}\}$. Prove that $\overline{L}$ is context free.

PROBLEM 5 (8 + 4 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) Prove that the language of $G = (V, \Sigma, R, S)$, where $V = \{S\}, \Sigma = \{a, b\}$, and $R = \{S \to SSS, S \to bS, S \to Sb, S \to a\}$ is

$$L(G) = \{w\text{: the number of a's in } w \text{ is odd}\}$$

Hint: To show $\subseteq$, proceed by induction on the length of the derivation. To show $\supseteq$, proceed by induction on the number of b's.

(B) Show that the grammar of part (A) is ambiguous but that there is an unambiguous grammar for L(G).

PROBLEM 6 (Challenge! 3 points)

Show that every context-free language over a unary alphabet is regular.