CS3133

MIDTERM EXAM

Name______________

Date: September 19, 2011

All nonelectronic documentation permitted

1. (25 points) Is $\{0^i 1^j \mid i \neq j\}$ regular? Justify your answer. (HINT: You might want to appeal to closure properties of regular languages.)

2. (25 points) Prove or give a counterexample to the following:

<u>CONJECTURE</u>: For any regular languages L_0 and L_1, the set of all strings with a prefix from L_0 and a suffix from L_1 and any string between them is regular.

3. (25 points) Describe an NFA (you may use ε-transitions) to accept the language

$$\left(1^{*}01^{*}0\right)^{*}1^{*}+\left(0+1\right)^{*}00\left(0+1\right)^{*} \quad .$$

4. (25 points) Give a context-free grammar without $\varepsilon -$ productions or unit productions to generate the set of strings from $aa^*(b+c)(b+c)^*$ with the number of *a*'s equal to the number of *b*'s plus the number of *c*'s, $\#(a)=\#(b)+\#(c)$. That is, the language contains a nonempty string of *a*'s followed by a string of *b*'s and *c*'s, with the sum of the numbers of *b*'s and *c*'s equal to the number of *a*'s. The language contains *aaabcb* and *aaaccc*, but it does not contain ε or *aaabc* or *aabcba*.

CS3133

Solutions to Midterm Exam

1. $\{0^i1^j \mid i \neq j\}$ is not regular. To show this, assume that $\{0^i1^j \mid i \neq j\}$ is regular. Because the regular languages are closed under complement, then $\sim\{0^i1^j \mid i \neq j\}$ would have to be regular. We now want to restrict our attention to strings of 0's followed by 1's. Because the regular languages are closed under intersection, $\left(\sim\{0^i1^j \mid i \neq j\}\right)\cap 0^*1^*$ must also be regular. But $\left(\sim\{0^i1^j \mid i \neq j\}\right)\cap 0^*1^* = \{0^i1^j \mid i = j\}$ which we showed to be false. So, by contradiction, $\{0^i1^j \mid i \neq j\}$ can not be regular.

2. The CONJECTURE is true. We show this in two ways (using NFAs and using regular expressions):

Given regular languages L_0 and L_1, there must be NFAs $M_0 = (Q_0, \Sigma, \Delta_0, S_0, F_0)$ and $M_1 = (Q_1, \Sigma, \Delta_1, S_1, F_1)$ to accept them. We can accept the set of all strings with a prefix from L_0 and a suffix from L_1 with the NFA with ε-transitions $M^* = (Q_0 \cup Q_1 \cup \{q\}, \Sigma, \Delta^*, S_0, F_1)$ where $q \notin Q_0 \cup Q_1$ and Δ^* behaves like Δ_0 in M_0 and Δ_1 in M_1 except that we add $\Delta^*(p,\varepsilon) = \{q\}$ for each $p \in F_0$, $\Delta^*(q,a) = \{q\}$ for each $a \in \Sigma$, and $\Delta^*(q,\varepsilon) = S_1$.

Given regular languages L_0 and L_1, there must be regular expressions α and β describing them. We can describe the set of all strings with a prefix from L_0 and a suffix from L_1 with the regular expression $\alpha(a_1 + \ldots + a_n)^*\beta$ where $\Sigma = \{a_1, \ldots, a_n\}$.

3. In the following, the p's are the states of an NFA to accept $(1^*01^*0)^*1^*$, and the q's are the states of an NFA to accept $(0+1)^*00(0+1)^*$. To accept the union of these languages, we start with an ε-transition to the start state of each. Our NFA with ε-transitions is $(\{s, p_0, p_1, p_2, q_0, q_1, q_2\}, \{0,1\}, \Delta, \{s\}, \{q_2, p_0, p_2\})$, where

Δ	0	1	ε
s	$\varnothing$	$\varnothing$	$\{p_0, q_0\}$
p_0	$\{p_1\}$	$\{p_0\}$	$\varnothing$
p_1	$\{p_2\}$	$\{p_1\}$	$\varnothing$
p_2	$\varnothing$	$\{p_0\}$	$\{p_0\}$
q_0	$\{q_0, q_1\}$	$\{q_0\}$	$\varnothing$
q_1	$\{q_2\}$	$\varnothing$	$\varnothing$
q_2	$\{q_2\}$	$\{q_2\}$	$\varnothing$

4.

$$S \rightarrow aSB \mid aB$$
$$B \rightarrow b \mid c$$