CONCORDIA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE & SOFTWARE ENGINEERING

COMP 335/4 Theoretical Computer Science Winter 2013

Assignment 3

Due date: March 22, 2013, by 11:59 p.m. EST

1. Use the product construction to find a DFA for the language

$$L\Big((\boldsymbol{ab}+\boldsymbol{ba})^*\Big) \cap L\Big((\boldsymbol{abb}+\boldsymbol{aab}+\boldsymbol{baa}+\boldsymbol{bba})^*\Big).$$

2. For a string $w = a_1a_2a_3a_4a_5a_6a_7\ldots$, define $third(w) = a_3a_6a_9\ldots$. Then, for a language L, define $third(L) = \{third(w) : w \in L\}$. Show that if L is regular, then $third(L)$ is also regular.

   Hint: Construct an ϵ-NFA from the DFA for L.

3. Let h be the homomorphism $h : \{a, b\} \to \{0, 1\}^*$ given by $h(a) = 01$, $h(b) = 011$, and define $L = \{w \in \{0, 1\}^* \ : \ n_1(w) \not\equiv 0 \ (\textbf{mod}\ 3)\}$. Construct a DFA for $h^{-1}(L)$.

4. Draw the table of distinguishabilities for the DFA below (run the TF algorithm), and then construct the minimum state equivalent DFA.

	0	1
$\to A$	B	E
B	C	F
$*\,C$	D	H
D	E	H
E	F	I
$*\,F$	G	B
G	H	B
H	I	C
$*\,I$	A	E

5. Find minimal DFA's for the following languages. In each case prove (!) that your DFA is minimal.

   (a) $\{a^nb^m : n \geq 2, m \geq 1\}$

   (b) $\{a^nb : n \geq 0\} \cup \{b^na : n \geq 1\}$

   (c) $\{a^n : n \geq 0, n \neq 3\}$

6. Design a CFG for each of the following languages

   (a) $L = \{a^nb^m : 0 \leq n \leq m \leq 2n\}$.

   (b) $L = \{a^ib^ja^k : i = j \text{ and } k \geq 0 \text{ or } i \geq 0 \text{ and } j > k\}$.

7. In each case, what language is generated by CFG's below. Justify your claim (prove it!)

   (a) $S \to aSa|bSb|aAb|bAa,\ A \to aAa|bAb|a|b|\epsilon$

   (b) $S \to aS|bS|a$

   (c) $S \to SS|bS|a$

   (d) $S \to SaS|b\ S \to aT|bT|\epsilon,\ T \to aS|bS.$

8. In each case below, show that the grammar is ambiguous, and find an equivalent unambiguous grammar.

   (a) $S \to SS|ab|a$

   (b) $S \to ABA,\ A \to aA|\epsilon,\ B \to bB|\epsilon$

   (c) $S \to aSb|aaSb|\epsilon$

9. Design a PDA to accept each of the following languages. You may design your PDA to accept either by final state or empty stack, whichever is more convenient.

   (a) The set of strings over $\{0, 1\}$ such that no prefix has more 1's than 0's.

   (b) The set of strings with twice as many 0's as 1's.

   (c) The set of strings over $\{a, b\}$ that are *not* of the form ww, that is, not equal to any string repeated.