T-79.1001/1002 **Autumn 2005**
Introduction to Theoretical Computer Science T/Y
Tutorial 2, 27–28 September
Problems

Remember to enroll for the course using the TOPI registration system by 23 September. Registration is compulsory.

Homework problems:

1. Let $\Sigma = \{a, b\}$. Give some examples of strings from each of the following languages (at least three strings per language):

   (a) $\{w \in \Sigma^* \mid w$ the number of a's in w is even and the number of b's is divisible of three$\}$;

   (b) $\{a^{2n}b^{3m} \mid n, m \geq 0\}$;

   (c) $\{uvu^Rv^R \mid u, v \in \Sigma^*\}$;[1]

   (d) $\{w \in \Sigma^* \mid \exists u, v \in \Sigma^*$ s.t. $w = uu = vvv\}$.

2. The *reversal* of a string $w \in \Sigma^*$, denoted w^R, is defined inductively by the rules:

   (i) $\varepsilon^R = \varepsilon$;

   (ii) if $w = ua$, where $u \in \Sigma^*$ and $a \in \Sigma$, then $w^R = au^R$.

   It was proved in class (cf. also Lewis & Papadimitriou, p. 43) that for any strings $u, v \in \Sigma^*$ it is the case that $(uv)^R = v^Ru^R$. Prove in a similar manner, by induction based on the above definition of reversal, the following facts:

   (a) $(w^R)^R = w$;

   (b) $(w^k)^R = (w^R)^k$, for any $k \geq 0$.

3. Design finite automata that recognise the following languages:

   (a) $\{w \in \{a, b\}^* \mid w$ contains ab as a substring$\}$;

   (b) $\{w \in \{a, b\}^* \mid w$ contains abb as a substring$\}$;

   (c) $\{w \in \{a, b\}^* \mid w$ does not contain abb as a substring$\}$;

   (d) $\{w \in \{a, b\}^* \mid ab$ occurs exactly twice as a substring in $w\}$;

   (e) $\{w \in \{0, 1\}^* \mid w$ contains an even number (possibly zero) of 0's$\}$;

   (f) $\{w \in \{0, 1\}^* \mid$ the number of 1's in w is divisible by three (or possibly zero)$\}$;

   (g) $\{w \in \{0, 1\}^* \mid w$ begins and ends with different symbols$\}$.

PLEASE TURN OVER

[1] For a definition of the notation w^R see Problem 2.

Demonstration problems:

4. Show that any alphabet Σ with at least two symbols is comparable to the binary alphabet $\Gamma = \{0, 1\}$, in the sense that strings over Σ can be easily encoded into strings over Γ and vice versa. How much can the length of a string change in your encoding? (I.e., if the length of a string $w \in \Sigma^*$ is $|w| = n$ symbols, what is the length of the corresponding string $w' \in \Gamma^*$?) Could you design a similar encoding if the target alphabet consisted of only *one* symbol, e.g. $\Gamma = \{1\}$?

5. Design finite automata that recognise the following languages:

   (a) $\{a^m b^n \mid m = n \mod 3\}$;

   (b) $\{w \in \{a, b\}^* \mid w$ contains equally many a's and b's, modulo $3\}$.

   (The notation "$m = n \mod 3$" means that the numbers m and n yield the same remainder when divided by three.)

6. Design a finite automaton that recognises sequences of integers separated by plus and minus signs (e.g. 11+20-9, -5+8). Implement your automaton as a computer program that also calculates the numerical value of the input expression.