Homework 6

Finite Automata on Infinite Words and Trees   Winter semester, 2009-2010

Exercise 1 (15 points)**.** Let $\Sigma = \{a, b, c\}$. Define Büchi tree automata which recognize the following tree languages:

1. the set of all trees t such that: for all positions p, if $t(p) = a$ then $t(p') = a$ for all $p \sqsubseteq p'$
2. the set of all trees in which every occurrence of a on a path is immediately followed by bb
3. the set of all trees in which every path contains at least one c

Exercise 2 (15 points)**.** Let $\Sigma = \{a, b\}$. Define Müller tree automata which recognize the following tree languages:

1. the set of all trees containing exactly two occurrences of b
2. the set of all trees all of whose paths have either finitely many a's or contain only a's
3. the set of all trees where for every position p, either $p1$ or $p2$ is labelled by a

Exercise 3 (10 points)**.** Let $\Sigma = \{a, b\}$. Define parity tree automata which recognize the following tree languages:

1. the set of all trees which have a path b^ω
2. the set of all trees whose paths all have finitely many occurrences of bb

Exercise 4 (15 points)**.** Prove that every Müller-recognizable tree language is recognized by a Müller automaton which is complete and contains a single initial state.

Exercise 5 (35 points)**.** The projection of a $\Sigma_1 \times \Sigma_2$-tree τ over the alphabet Σ_1 is the tree $pr_1(\tau)$ defined such that $pr_1(\tau)(v) = q_1$ when $\tau(v) = (q_1, q_2)$. The projection of a language L of $\Sigma_1 \times \Sigma_2$-trees onto Σ_1 is the set $\{pr_1(\tau) \mid \tau \in L\}$ of Σ_1-trees.

Prove that the set of Müller-recognizable tree languages is closed under union, intersection, and projection.

Is the set of Büchi-recognizable tree languages also closed under these operations?

Exercise 6 (10 points)**.** Explain briefly why the transformation from Müller word automata into equivalent Büchi word automata from Chapter 2 cannot be used to transform Müller tree automata into equivalent Büchi tree automata.