CS 373: Theory of Computation
Sariel Har-Peled and Madhusudan Parthasarathy

Problem Set 6

Due: Thursday, March 19, 2009 at 12:30 in class (i.e., SC 1105)
Version: **1.01**

(Q1) Building Recursive Automata.
[**Category**: Construction, **Points**: 10]

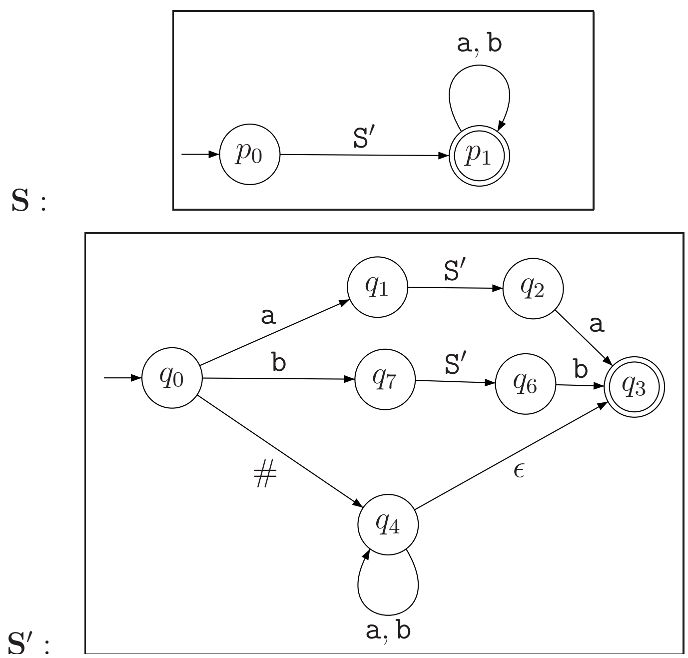
For each of the following languages construct a recursive automaton for it, and briefly describe why it works. Also, for each of these languages, pick a word of length at least 6 in the language and show the run of your automaton on it, including stack contents at each point of the run:

(a) $L_1 = \left\{ \mathtt{a}^i \mathtt{b}^j \;\middle|\; i \geq 0, 3i \geq j \geq 2i \right\}$ over the alphabet $\{\mathtt{a}, \mathtt{b}\}$.

(b) $L_2 = \left\{ w_1 \$ w_2 \$ \mathtt{0}^i \$ \mathtt{1}^j \;\middle|\; w \in \{\mathtt{a}, \mathtt{b}\}^*, i = |w_2| \text{ and } j = |w_1| \right\}$
Here L_2 is defined over the alphabet $\{\mathtt{a}, \mathtt{b}, \$, \mathtt{0}, \mathtt{1}\}$.

(Q2) Understanding Recursive Automata.
[**Category**: Understanding, **Points**: 10]

For the following recursive automaton with initial module S, give the language of the automata precisely.



(Q3) CYK Parsing.
[**Category**: Understanding, **Points**: 10]

For the following CNF grammar (with start symbol S) and the following string:

book the flight through champaign

(assume spaces differentiate the various terminals and nonterminals [not single characters]), generate a valid parse tree using the CYK parsing algorithm. Turn in the filled out chart (matrix) as well.

$S \rightarrow$ `NP VP` | `X2 PP` | `VERB NP` | `VERB PP` | `VP PP` | *book*

`NP` $\rightarrow$ *i* | *she* | *me* | *champaign* | `DET NOMINAL`

`DET` $\rightarrow$ *the* | *a*

`NOMINAL` $\rightarrow$ *book* | *flight* | *meal* | `NOMINAL NOUN` | `NOMINAL PP`

`NOUN` $\rightarrow$ *book* | *flight* | *champaign*

`VP` $\rightarrow$ *book* | *include* | *prefer* | `VERB NP` | `X2 PP` | `VERB PP` | `VP PP`

`VERB` $\rightarrow$ *book* | *flight*

`X2` $\rightarrow$ `VERB PP`

`PP` $\rightarrow$ `PREPOSITION NP`

`PREPOSITION` $\rightarrow$ *through*

In the above, the terminals are
$\Sigma = \{book, i, she, me, champaign, the, a, flight, meal, include, prefer, through\}$.

(Q4) Closure Properties of CFLs
[**Category**: Proof, **Points**: 10]

For each of the following languages, prove that they are not context free using closure properties discussed in class to generate a known non-context free language such as $\{a^n b^n c^n \mid n \geq 0\}$. You can assume that CFLs are closed under homomorphisms and inverse homomorphisms as well.

(a) $L_1 = \{w \in \{a,b,c,d\}^* \mid \#_a(w) = \#_b(w) = \#_c(w) = \#_d(w)\}$ where $\#_a(w)$ denotes the number of a's in w, etc.

(b) $L_2 = \{0^i \# 0^{2i} \# 0^{3i} \mid i \geq 0\}$

(Q5) Shu□e
[**Category**: Proof, **Points**: 10]

For a given language, L, we will define Shuffle(L) as follows:
$\{w \mid y \in L,\ |y| = |w| \text{ and, } w \text{ is a permutation of letters in } y\}$
For instance if $L = \{ab, ada\}$, Shuffle(L) $= \{ab, ba, aad, ada, daa\}$
Prove that if L is a regular language, then Shuffle(L) is not necessarily a CFL. In other words, prove that the statement "For every regular language L, Shuffle(L) is a CFL" is false.