CS331 Assignment #2

Due Date: Nov 17, 2003

Be brief, complete and stick to what has been asked. If needed, you may cite results/proofs covered in class without reproducing them.

Do not copy solutions from others
Penalty for copying: FR grade

1. *[5+10 pts]* Let $G = (\{S, A, B\}, \{0, 1\}, P, S)$ be a CFG where P is given by:

$$\begin{array}{lclclcl} S & \rightarrow & ASB & \mid & 10 & & \\ A & \rightarrow & BA & \mid & ASA & \mid & 0 \\ B & \rightarrow & AB & \mid & 1 & & \end{array}$$

   (a) Convert G to an equivalent grammar G' (i.e., G' generates the same language as G) in which each production rule is in one of the following two forms:

   - $X \rightarrow YZ$, where X, Y, Z are non-terminals, and X is *distinct* from Y or Z
   - $X \rightarrow 0$ or $X \rightarrow 1$.

   (b) Define $graft(L)$ as $\{w \mid w = x.y.z,\ x, y, z \in (\mathbf{0} + \mathbf{1})^*,\ x.z \in L \text{ and } y \in L\}$. Thus a string in $graft(L)$ is formed by taking a string in L, splitting it into two parts and grafting another string from L in between. Give a CFG for $graft(L(G))$, where G is the grammar given at the beginning of the question.

2. *[7.5 + 5 + 7.5 points] This question is intended to test your skills in constructing CFGs. It is not meant to test your skills in constructing PDAs and then mechanically converting them to CFGs. Therefore, you* ***must not*** *construct a PDA and then mechanically transform it to a CFG for purposes of this question. Any such attempt will fetch no points.*

   Let $L = \{w \mid w \in (\mathbf{0} + \mathbf{1})^+, N_0(w) \leq N_1(w) \leq 2N_0(w)\}$, where $N_i(w)$ represents the number of is in string w.

(a) Give a CFG for L.

(b) Is $L^c = \Sigma^* - L$ context-free?

(c) Let $\frac{1}{2}L = \{x \mid \exists y \in L,\ |x| = |y|,\ x.y \in L\}$. Is $\frac{1}{2}L$ context-free?

In parts (b) and (c), you must give a CFG *(not a PDA)* if your answer is in the affirmative. Otherwise, you must prove that the corresponding language is not context-free.

3. *[15 points]* Show that there exist context-free languages that cannot be recognized by any PDA with 3 states and 4 stack symbols, by empty stack. You must give a context-free language and prove why no PDA with the above number of states and stack symbols can accept your language by emptying its stack.

4. *[7.5 + 7.5 points]* Let L_1 and L_2 be languages over the alphabet Σ. Define the *right quotient* of L_2 by L_1 as $L_1 \backslash L_2 = \{w \mid w \in \Sigma^*,\ \exists y \in L_1 \text{ and } y.w \in L_2\}$.

   (a) If L_1 is regular and L_2 is context-free, is $L_1 \backslash L_2$ necessarily context-free?

   (b) If L_1 and L_2 are recursively enumerable, is $L_1 \backslash L_2$ also recursively enumerable?

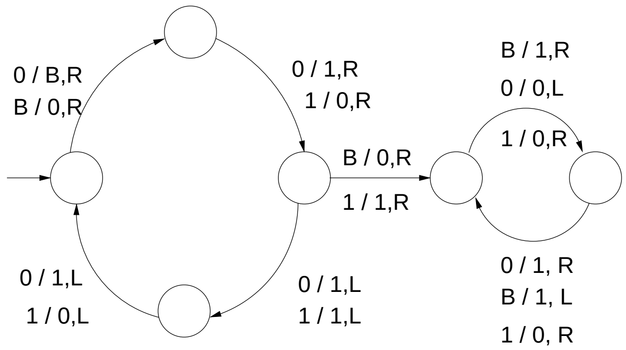
   In each case, you must either give a proof or a counterexample.

5. *[5 + 5 + 10 points]* Let $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ be a Turing Machine. Define $I(M)$ as $\{w \mid w \in \Sigma^*, \exists^\omega i \text{ such that } q_0\ w \overset{i}{\vdash} \alpha_i\ q_0\ \beta_i\}$. Here, $\exists^\omega i$ means "there exist infinitely many is". Thus, $I(M)$ is the set of strings that cause M to visit its starting state infinitely often. If M is a non-deterministic machine, $I(M)$ contains all strings in Σ^* that lead to at least one sequence of IDs (instantaneous descriptions) in which the starting state is visited infinitely often.

   Consider the Turing Machine shown in Fig. 1. Let $H(M)$ be the language accepted by the machine by halting.

   (a) Construct a Turing Machine M' such that $I(M') = H(M)$.

   (b) Construct a Turing Machine M'' such that $I(M) = H(M'')$. *[Hint: Don't attempt a general construction. Exploit properties of the given machine M.*



$\Sigma = \{0, 1\}$
$\Gamma = \{0,1,B\}$

Figure 1:

(c) Show that there does not exist any algorithm that takes a Turing Machine M_1 and a string x_1 as inputs, and halts by producing

- "YES" if M_1 run with string x_1 on its tape visits its starting state infinitely often.
- "NO" otherwise.