CSCI 510 **Models of Computation**

HW8: One Way or Another

Due: Nov 17, 2010

Include your *full name*, *CS login*, and the problem number(s) on each piece of paper you hand in, and please staple your pages together before handing in.

While collaboration is encouraged in this class, please remember not to take away notes from collaboration sessions other than your scheduled lab section.

Problem 1

Describe pushdown automata for the language $L = \{0^r 1^s \mid r \leq s \leq 2r\}$. An English description of what the PDA pushes onto the stack when and what it pops off when is completely acceptable.

Problem 2

Let $G = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathrm{s})$ be the context-free grammar defined below:

$$\begin{aligned} \mathcal{N} &= \{\mathbf{S}\} \\ \mathcal{T} &= \{a, b, 0\} \\ \mathcal{R} &= \{\mathbf{S} \to 0, \mathbf{S} \to \mathbf{SS}, \mathbf{S} \to a\mathbf{S}b\} \\ \mathrm{s} &= \mathbf{S}. \end{aligned}$$

Generate a parse tree for the string $a0b0$ using the above rules. Please show each intermediate step in your solution.

The following questions are lab problems. Please remember to prepare a solution for your assigned problem before going to your lab section.

Lab Problem 1

Show that the following language $L = \{ww \mid w \in \{0, 1\}^*\}$ is not context-free.

Hint: The intersection of a context-free language and a regular language is context-free.

Lab Problem 2

Consider the following language L:

$$L = \Big\{ s_1\#s_2 \quad | \quad s_1 \in \{ \begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}0\\1\end{bmatrix}, \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}1\\1\end{bmatrix} \}^*, s_2 \in \{0,1\}^*,$$
$$\text{and if } s_1 = \begin{bmatrix}x_0\\y_0\end{bmatrix} \begin{bmatrix}x_1\\y_1\end{bmatrix} \cdots \begin{bmatrix}x_n\\y_n\end{bmatrix}, \text{ then } s_2 = x + y$$
$$\text{where } x = \Sigma_{i=0}^{n} x_i 2^i \text{ and } , y = \Sigma_{i=0}^{n} y_i 2^i \Big\}$$

In other words, for a string $s_1\#s_2$, we interpret $s_1 = \begin{bmatrix}x\\y\end{bmatrix}$ as two binary integers x, y with least-significant bit first, and s_2 as a binary integer with most-significant bit first. $s_1\#s_2 \in L$ whenever $s_2 = x + y$ (i.e., the sum of the two numbers represented as pairs equals the number after the #).

To ensure that all additions have a unique representation, we require the following:

- $s_2 \in \varepsilon \cup 1\{0,1\}^*$ (s_2 either is empty or begins with a 1.)
- $s_1 \in \varepsilon \cup \{ \begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}0\\1\end{bmatrix}, \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}1\\1\end{bmatrix} \}^* \{ \begin{bmatrix}0\\1\end{bmatrix}, \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}1\\1\end{bmatrix} \}$
  (x and y cannot both begin with 0s.)

Some examples to help you understand the definition of the language:

- $\# \in L$
- $\begin{bmatrix}1\\1\end{bmatrix} \begin{bmatrix}0\\1\end{bmatrix} \begin{bmatrix}1\\0\end{bmatrix} \#1000 \in L$ because $101 + 011 = 1000$
- $\begin{bmatrix}1\\1\end{bmatrix} \begin{bmatrix}0\\1\end{bmatrix} \begin{bmatrix}0\\0\end{bmatrix} \#100 \notin L$ because x and y both begin with extraneous 0s.
- $\begin{bmatrix}1\\1\end{bmatrix} \begin{bmatrix}0\\1\end{bmatrix} \begin{bmatrix}1\\0\end{bmatrix} \#01000 \notin L$ because s_2 begins with extraneous 0s.
- $\begin{bmatrix}1\\1\end{bmatrix} \begin{bmatrix}0\\1\end{bmatrix} \begin{bmatrix}1\\0\end{bmatrix} \#1001 \notin L$ because $101 + 011 \neq 1001$

a. Prove that L is not regular.

b. Design a CFG for L.

Lab Problem 3

Throughout this problem, $L_{01} = \{0^n 1^n \mid n \geq 0\}$.

a. Consider a Turing Machine T that does the following:

- T takes as input $\langle M, w \rangle$—a description of a Turing Machine M[1] and a string w.
- T prints to its tape as output a description of a new machine M' where $L(M') = L_{01}$ if M accepts w, and $L(M') = \emptyset$ otherwise.

Give an English description of what such a T would do to produce the desired output from input $\langle M, w \rangle$, and explain why the M' described in its output satisfies the desired properties.

Hint: Keep in mind that since T is just printing a *description* of another Turing Machine it will always halt, whereas M' does not always have to.

b. Define the language A_{TM} as follows:

$$A_{TM} = \{\langle M, w \rangle \mid M \text{ is a Turing machine that accepts } w\}$$

Note that A_{TM} is undecidable. Using part (a) and this fact, show that the language $B = \{\langle M \rangle \mid L(M) = L_{01}\}$ is not decidable. You need to show that if B is decidable, then so is A_{TM}. In order to do this, suppose that there exists a Turing machine R that decides B. Then build a Turing machine that decides A_{TM}, using as subroutines both R and the Turing machine you designed in part (a).

[1] A description of a TM describes the operations of the TM in a way that allows another TM to simulate the described TM.