A GIRL HAS NO NAME
Computer Science 81
Homework 12
4/25/17

Notation: If x is a string and σ is a symbol, then $\#_\sigma(x)$ is the number of times that σ occurs in x. For example, $\#_1(01101) = 3$. Also $|x|$ is the total number of symbols in x.

1, 2, 3, 4

1. [3 points] Give a context-free grammar that generates the following language over terminal alphabet $\{1, +, =\}$:

$$L = \left\{1^m + 1^n = 1^{n+m} \,|\, m > 0 \wedge n > 0\right\}$$

The first few strings in L are:

$$\begin{aligned}
&1 + 1 = 11\\
&1 + 11 = 111\\
&11 + 1 = 111\\
&11 + 11 = 1111\\
&11 + 111 = 11111\\
&111 + 11 = 11111
\end{aligned}$$

■

2. [7 points] Consider the following context-free grammar G, with terminal alphabet $\{0,1\}$ and start symbol S. The productions are:

$$S \to \epsilon \mid 0S1S \mid 1S0S$$

It is claimed that this grammar generates the language

$$L = \{x \in \{0,1\} * \mid \#_0(x) = \#_1(x)\}.$$

a. [2 points] Prove that $L(G) \subseteq L$. For this part, it is natural to use mathematical induction on the number of steps in the derivation of a string from S.

b. [5 points] Prove that $L \in L(G)$. For this part, you would probably use strong mathematical induction on the length of strings in the language.

   That is, given $x \in L$ to show $x \in L(G)$, make the inductive assumption that for every y with $|y| < |x|, y \in L$ implies $y \in L(G)$. Consider breaking x into smaller pieces on which the induction hypothesis can be used, then show how the pieces are put together using the productions.

   As an example, consider a string such as 1100010011. How would you break it down?

■

3. [6 points] Determine for each of the following grammars whether or not it generates an infinite language. In each case the start symbol is S and the terminal alphabet is $\{0, 1, 2\}$. Explain your reasoning.

a. $S \to A \mid B \qquad A \to A0 \qquad B \to B1 \qquad C \to S \mid C2 \mid \epsilon$

b. $S \to A0 \qquad A \to 1B \qquad B \to A0 \mid 2$

c. $S \to 0AB1 \mid \epsilon \qquad A \to B1 \qquad B \to A2$

■

4. [Optional Extra Credit: 15 points] We know that it is undecidable whether the language of a Turing machine is infinite. For context-free grammars, however, this property *is* decidable. Show this by giving an algorithm for deciding whether the language of a context-free grammar is infinite. Your write-up should be an algorithm in pseudocode, or better, an actual implementation.

■