CS331 Practice Problem Set #2

1. Let $\Sigma = \{0, 1\}$. Give a CFG that generates all strings of the form wx, where $w, x \in \Sigma^+$, $|w| = |x|$ and w is *not* the reverse of x. Give an intuitive description of the set of strings derived from each non-terminal.

2. Let $\circ$ be a non-associative binary operator that evaluates from right to left. Thus, $a \circ b \circ c = a \circ (b \circ c) \neq (a \circ b) \circ c$. Parentheses can be used to modify the order of evaluation. Consider the grammar $G = (\{S\}, \{\mathbf{id}, \circ, (,)\}, P, S)$ for expressions involving $\circ$, where P is given by:

$$S \rightarrow \mathbf{id} \mid (S) \mid S \circ S$$

Clearly, this is an ambiguous grammar. Your task is to design an umambiguous grammar $G' = (V', \Sigma, P', S')$ such that:

- $L(G) = L(G')$.
- Every production rule in P' has one of the following forms:
  - $A \rightarrow \mathbf{id}$, or
  - $A \rightarrow (B)$ for $B \in V'$, or
  - $A \rightarrow B \circ C$ for $B, C \in V' \cup \{\mathbf{id}\}$.
- For every expression e, the derivation tree for e is such that the correct value of e is obtained by "evaluating" the tree in the following manner:
  Evaluating derivation tree for e:
  - The values of leaf nodes labeled **id** are assumed known.
  - The value of a non-leaf node representing the production $A \rightarrow \mathbf{id}$ is the value of child **id**.
  - The value of a non-leaf node representing the production $A \rightarrow (B)$ is the value of child B.
  - The value of a nonleaf node representing $A \rightarrow B \circ C$ is the (value of child B) $\circ$ (value of child C).

– The value of the root node is the value of e.

For example, G' should give Tree b and not Tree a as the derivation tree for $\mathbf{id} \circ (\mathbf{id} \circ \mathbf{id}) \circ \mathbf{id} \circ \mathbf{id} \circ \mathbf{id}$.

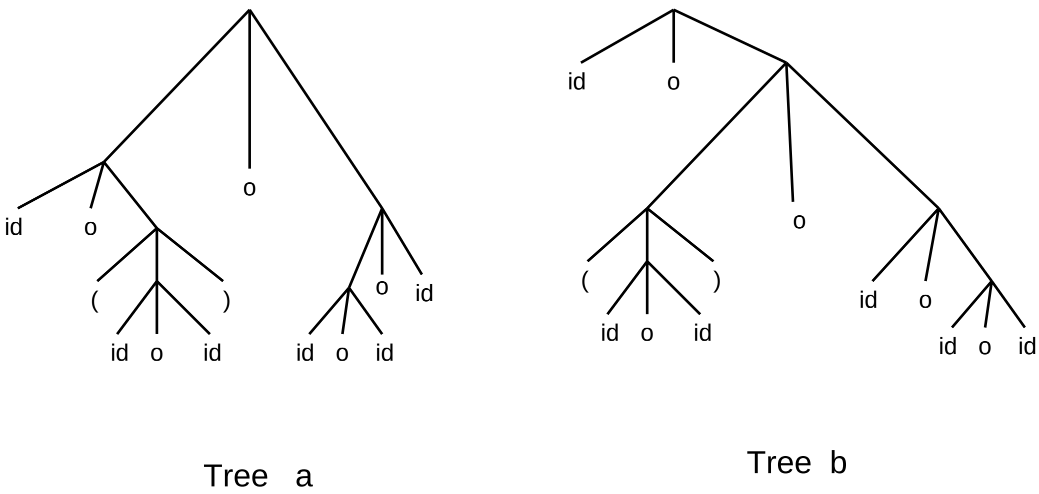

Figure 1:

3. Design a PDA that accepts by empty stack all strings with equal no. of 0s and 1s. The 0s and 1s can appear in any order and may be interspersed between each other. Convert this PDA to a CFG for the same language. Is the grammar ambiguous?

4. Let a be a symbol in Σ. For $L \subseteq \Sigma^*$, define $a\backslash L = \{w \in \Sigma^* \mid aw \in L\}$. Show that if L is context-free, so is $a\backslash L$.

5. Which of the following languages are CFLs? Justify your answer.

   (a) $\{0^i 1^j 2^k \mid i < j < k\}$.

   (b) $\{w \in (\mathbf{0} + \mathbf{1} + \mathbf{2})^* \mid w$ does not contain the same number of all 3 symbols $\}$.

   (c) $\{uawb \mid u, w \in (\mathbf{a} + \mathbf{b})^*,\ |u| = |w|\}$.

   (d) $\{b_i \# b_{i+1} \mid b_i$ is i in binary, $i \geq 1\}$. The alphabet here is $\{0, 1, \#\}$.

6. Let L be a CFL and R a regular language. Is $R - L$ context-free? If not, give a counterexample, else give a proof of context-freeness.

7. You are given a PDA that accepts L by empty stack. How would you construct another PDA that accepts L^R (all strings of L reversed) by empty stack?