Solutions for the selected exercises from E5

25 - Feb - 2020

5.1.1) (a) $\{0^n 1^n \mid n \geq 1\}$

$$S \rightarrow 01 \mid 0S1$$

5.1.1) (b) $\{a^i b^j c^k \mid i \neq j \text{ or } j \neq k\}$

$S \rightarrow AB \mid CD$ // $AB: j \neq k$, $CD: i \neq j$

$A \rightarrow aA \mid \varepsilon$ // a^*

$B \rightarrow bBc \mid E \mid cD$ // b^*c^* s.t. $\#b \neq \#c$

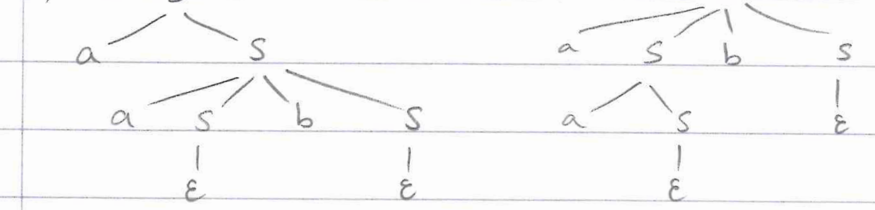
$C \rightarrow aCb \mid E \mid aA$ // a^*b^* s.t. $\#a \neq \#b$

$D \rightarrow cD \mid \varepsilon$ // c^*

$E \rightarrow bE \mid b$ // b^+

5.4.1) $S \rightarrow aS \mid aSbS \mid \varepsilon$ "aab"

(a) parse trees



(b) Leftmost derivations:

(I) $S \Rightarrow aS \Rightarrow aa\overset{(1)}{S}b\overset{(2)}{S} \Rightarrow aabS \Rightarrow aab$

(II) $S \Rightarrow a\underset{(1)}{S}b\underset{(2)}{S} \Rightarrow aa\underset{(1)}{S}b\underset{(2)}{S} \Rightarrow aabS \Rightarrow aab$

(c) Rightmost derivations:

(I) $S \Rightarrow aS \Rightarrow aa\underset{(2)}{S}b\underset{(1)}{S} \Rightarrow aaSb \Rightarrow aab$

(II) $S \Rightarrow a\underset{(2)}{S}b\underset{(1)}{S} \Rightarrow aSb \Rightarrow aaSb \Rightarrow aab$

5.4.2) Property ≜ every prefix has at least as many a's as b's.

(I) Proof by induction on natural numbers:

$|w| = 0 \rightarrow w = \varepsilon \Rightarrow [S \Rightarrow \varepsilon]$ ✓

$|w| = 1 \rightarrow w = a \Rightarrow [S \Rightarrow aS \Rightarrow a]$ ✓

$|w| = 2 \rightarrow w = aa \Rightarrow [S \Rightarrow aS \Rightarrow aaS \Rightarrow aa]$ ✓

$\hookrightarrow w = ab \Rightarrow [S \Rightarrow aSbS \Rightarrow abS \Rightarrow ab]$ ✓

⋮

I.H. $|w| = k-1 \Rightarrow |a_w| \geq |b_w|$

I.H. $|w| = k \Rightarrow |a|_w \geq |b|_w$

I.J. $|w| = k+1 \Rightarrow$ according to the given production rules, it can only be constructed by prepending an 'a', and possibly adding one 'b' to the string.

Therefore, the property holds for all lengths.

(II) Proof by structural induction:

$p(\varepsilon): |a_\varepsilon| \geq |b_\varepsilon| \Rightarrow 0 \geq 0$ ✓

$p(S) \Rightarrow p(aS)$

$|a_S| \geq |b_S| \Rightarrow |a_S|+1 \geq |b_S| \Rightarrow |a_{aS}| \geq |b_{aS}|$ ✓

$p(S_1) \wedge p(S_2) \Rightarrow p(aS_1bS_2)$

$|a_{S_1}| \geq |b_{S_1}| \wedge |a_{S_2}| \geq |b_{S_2}| \Rightarrow$

$|a_{S_1}| + |a_{S_2}| \geq |b_{S_1}| + |b_{S_2}| \Rightarrow$

$|a_{S_1}| + |a_{S_2}| + 1 \geq |b_{S_1}| + |b_{S_2}| + 1 \Rightarrow |a_{aS_1bS_2}| \geq |b_{aS_1bS_2}|$ ✓

(I) shows all strings which satisfies the condition can be generated by the ~~grammar~~.

(II) proves the grammar only the strings such that the condition satisfies.

I, II $\Rightarrow$ the proof of the theorem!

5.4.3) (Similar to "else" dangling problem)

$$S \rightarrow aS \mid aTbS \mid \varepsilon$$
$$T \rightarrow aTbT \mid \varepsilon$$

The idea is to create a hierarchy in the grammar by introducing another nonterminal (T) that only can generate balanced a's and b's.

5.1.3) Prove that every regular language is a context-free language.

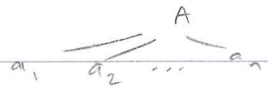
Proof by Construction:

Let A be a regular language. Then, there exists a DFA $N = (Q, \Sigma, \delta, q_0, F)$ such that $L(N) = A$. Build a context-free grammar $G = (V, \Sigma, R, S)$ as follows. Set $V = \{R_i \mid q_i \in Q\}$, i.e., G has a variable for every state of N. For every transition $\delta(q_i, a) = q_j$, add a production rule $R_i \rightarrow aR_j$. For every accepting state $q_i \in F$, add a rule $R_i \rightarrow \varepsilon$. Finally, make the start variable $S = R_0$.

5.2.2) Proof by induction on the derivation steps (m)

$|w| = n \qquad w = a_1 a_2 \cdots a_n$

Basis: $(m=1)$: Then we must have used a production $A \to w$ (or $A \to a_1 a_2 \cdots a_n$). The corresponding parse tree:

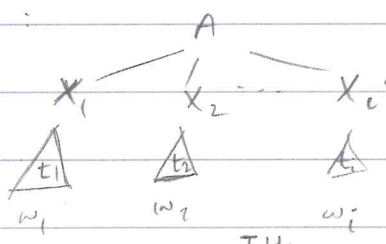


$\#\text{nodes} = |w| + 1 = n + m$ ✓

Induction step: w is infered in $m+1$ derivation steps.

Suppose that the last step was based on a production $A \to X_1 X_2 \cdots X_i$. We break w up as $w_1 w_2 \cdots w_i$, where $w_i = X_i$ when $X_i \in T$, and when $X_i \in V$ then w_i was previously inferred in $\mathcal{L}(X_i)$ in at most m steps.

By I.H., there are parse trees i with root X_i and yield w_i. Then, the following is a parse tree for w:



$\#\text{nodes} = 1 + \sum_i \#\text{nodes}(t_i) \overset{\text{I.H.}}{=} 1 + \sum_i (m_i + n_i) =$

$$1 + \sum_i m_i + \sum_i n_i = 1 + m + n = 1 + m + |w| \checkmark$$

$\downarrow$

with assumption that there is no epsilon on the right hand of any rule.