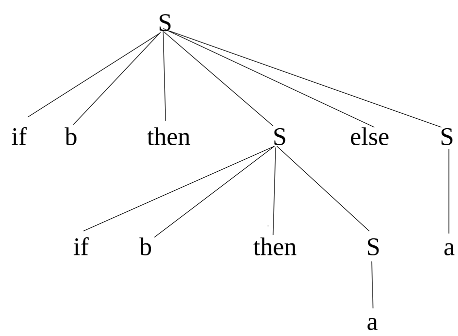
Memo: Tutorial 4

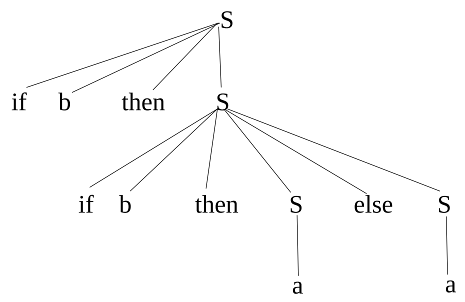
March 3, 2005

Question 1:

Two leftmost derivations:
S⇒ if b then S else S
⇒ if b then if b then S else S (replace the first S with if b then S)
⇒ if b then if b then a else S (replace the first S with a)
⇒ if b then if b then a else a (replace the S with a)

S⇒ if b then S
⇒ if b then if b then S else S (replace S with if b then S else S)
⇒ if b then if b then a else S (replace the first S a)
⇒ if b then if b then a else a (replace S with a)





Two different parse trees corresponding to if b then if b then a else a

Question 2:

Grammar: $S \to aSa \mid B \mid \varepsilon$
$B \to bB \mid \varepsilon$

Now convert this grammar to Chomsky normal form. First we add a new start variable:
$S_1 \to S$
$S \to aSa \mid B \mid \varepsilon$
$B \to bB \mid \varepsilon$

Remove now the ε rules. Remove $S \to \varepsilon$
$S_1 \to S \mid \varepsilon$
$S \to aa \mid aSa \mid B$
$B \to bB \mid \varepsilon$

Remove $B \to \varepsilon$
$S_1 \to S \mid \varepsilon$
$S \to aa \mid aSa \mid B$
$B \to b \mid bB$

Now we remove unit rules. Remove $S_1 \to S$
$S_1 \to aa \mid aSa \mid B \mid \varepsilon$
$S \to aa \mid aSa \mid B$
$B \to b \mid bB$

Remove $S_1 \to B$
$S_1 \to aa \mid aSa \mid b \mid bB \mid \varepsilon$
$S \to aa \mid aSa \mid B$
$B \to b \mid bB$

Remove $S \to B$
$S_1 \to aa \mid aSa \mid b \mid bB \mid \varepsilon$
$S \to aa \mid aSa \mid b \mid bB$
$B \to b \mid bB$

Convert remaining rules in proper from:
$S_1 \to aa \mid AC \mid b \mid DB \mid \varepsilon$
$S \to aa \mid AC \mid b \mid DB$
$B \to b \mid DB$
$C \to SA$
$A \to a$
$D \to b$

Question 3:

| a | a | b |

0 1 2 3

0			
A	1		
Ø	A	2	
S,B	S,B	B	3

Since we have S here we can derive aab from S in the given grammar.

Question 4:

| b | a | b | a |

0 1 2 3 4

0				
T	1			
R,T	R	2		
S	S	T	3	
S,R,T	S	R,T	R	4

Since we have S here the string baba can be derived from S in the given grammar.

Question 5: (Sipser, exercise 2.18(b))
Suppose $L = \{0^n\#0^{2n}\#0^{3n} | n \geq 0\}$ is context free and let p be the pumping length. Let $s = 0^p\#0^{2p}\#0^{3p}$. Then $|s| = 6p + 2 \geq p$, so according to the pumping lemma $s = uvwxy$ such that:

1. $uv^iwx^iy \in L$ for each $i \geq 0$
2. $|vx| > 0$
3. $|vwx| \leq p$

Notice that v or x can not contain a #, since then uv^2xy^2z will have too many #'s. We can thus assume that $v \neq \#$ and $x \neq \#$.
If $v = \varepsilon$ or $x = \varepsilon$ (by 2 v and x can't both equal ε), $uv^2wx^2y = 0^i\#0^j\#0^k$ and $i > p, j = 2p, k = 3p$ or $i = p, j > 2p, k = 3p$ or $i = p, j = 2p, k > 3p$. We can thus assume that $v \neq \varepsilon$ and $x \neq \varepsilon$.

It thus follows from 3) that we have the following possibilities for v and x:

- v and x is contained in the 0's before the first # - but then $uv^2wx^2y = 0^i\#0^{2p}\#0^{3p}$ and $i > p$, which is not in L.
- v is contained in the 0's before the first # and x is contained in the 0's between the first and second # - but then $uv^2wx^2y = 0^i\#0^j\#0^{3p}$ with $i > p$ and $j > 2p$, which is not in L.
- Similarly, if v and x is contained between the first and second # or after the second #, or if v is between the first and second # and x after the second # we conclude that $uv^2wx^2y \notin L$.

We conclude that it is not possible to divide s in five pieces with properties $1) - 3)$. Thus L is not regular.

Question 6: (Sipser, exercise 2.18(c))
Suppose $L = \{r\#s | r$ is a substring of s where $r, s \in \{a, b\}^*\}$ is context free and let p be the pumping length. Let $s = a^pb^p\#a^pb^p$. Then $|s| = 4p + 1 \geq p$, so according to the pumping lemma $s = uvwxy$ such that:

1. $uv^iwx^iy \in L$ for each $i \geq 0$
2. $|vx| > 0$
3. $|vwx| \leq p$

Notice that v or x can not contain a #, since then uv^2xy^2z will have too many #'s. We can thus assume that $v \neq \#$ and $x \neq \#$.
If $v = \varepsilon$, then $x \neq \varepsilon$ (by 2 not both v and x equal ε). Thus if x is before the #, $uv^2wx^2y = r\#s$ with $|r| > |s|$. Similarly, if x is after the #, $uv^0wx^0y = r\#s$

with $|r| > |s|$. For $v \neq \varepsilon$ and $x = \varepsilon$ we obtain similarly strings of the form uv^iwx^iy that is not in L. We thus assume that $v \neq \varepsilon$ and $x \neq \varepsilon$.

It follows from 3) that we have the following possibilities for v and x:

- vwx appears before the # - but then $uv^2wx^2y \notin L$, since $uv^2wx^2y = r\#s$ and $|r| > |s|$.
- Similarly, if vwx appears after the #, $uv^0wx^0y \notin L$, since $uv^0wx^0y = r\#x$ and $|r| > |s|$.
- v appears before the # and x after the #. Then v consits only of b's and x on of a's. But then $uv^2wx^2y = a^pb^i\#a^jb^p$ and $i, j > p$, thus $uv^2wx^2y \notin L$.

We conclude that it is not possible to divide s in five pieces with properties $1) - 3)$. Thus L is not regular.

Question 7: (Sipser, exercise 2.15)

CFL's are closed under union:
Let G_1 be a context-free grammar that generates the CFL L_1 and G_2 be a context-free grammar that generates the CFL L_2. Assume that S_1 is the start variable for L_1 and S_2 the start variable for L_2. Also assume that G_1 and G_2 have no variables in common. We can obtain a context grammar G_3 that generates $L_1 \cup L_2$ as follows:
The variables for G_3 are the union of the variables for G_1 and G_2, but we add a new start variable S_3. The rules are the union of the rules of G_1 and G_2, but we add the new rule $S_3 \rightarrow S_1 \mid S_2$.

CFL's are closed under concatenation:
Same as for union, but we add the rule $S_3 \rightarrow S_1S_2$ instead of $S_3 \rightarrow S_1 \mid S_2$.

CFL's are closed under star:
Let G_1 be a context-free grammar that generates the CFL L_1. Assume that S_1 is the start variable for L_1. We can obtain a context grammar G_3 that generates L_1^* as follows:
The variables for G_3 are the same as the variables for G_1. The rules are the same but we add the rule $S_1 \rightarrow S_1S_1 \mid \varepsilon$.

Question 8: (Sipser, exercise 2.16)
We use Definition 1.26 (the definition of a regular expression) on p64 of Sipser.

1. The regular expression a with $a \in \Sigma$ has an equivalent CFL generated by the grammar $S \rightarrow a$.
2. The regular expression ε has an equivalent CFL generated by the grammar $S \rightarrow \varepsilon$.
3. The regular expression $\emptyset$ has an equivalent CFL generated by the grammar $S \rightarrow S$.
4. If R_1 and R_2 are regulare expressions, and we have CFL's with grammars G_1 and G_2 that are respectively equivalent to R_1 and R_2, we showed in the previous question how to get a grammar for the union of the two languages.
5. We handle concatenation and star similar to union.

Question 9: (Sipser, exercise 2.19)
We may assume that in the derivation of w we first obtain all the necessary n variables, and then in the last n steps we replace the n variables by n terminals. Notice that since rules with varaibles on the the right hand side are of the form $A \rightarrow BC$, we need $(n-1)$ steps to obtain the n variables. Thus in total we need $n + (n-1) = 2n - 1$ steps.