5/16(水)　Right Parser
Deterministic Parser

제13의 물결
앨빈 토플러
1980.

Left Parser
" Produce-shift Parser
guess verify
Right Parser

$SLL(k) \subsetneq LL(k)$

$\subseteq LR(0) \subsetneq SLR(k) \subsetneq LALR(k) \subsetneq LR(k)$

~~reduce~~ shift-reduce (store & verify)

Parser (PDT) vs Grammar (cfg, lm/rm)

$x \in \Sigma^*$

$\theta \in \Gamma^*$　　$\pi \in P^*$

produce (guess)　$\tau(p) = r \in P$
shift (verify)　$\tau(s) = \varepsilon$

$|\theta| = |\pi| + |x|$
　　produce　shift

Thm 5.16 produce-shift parser is a left parser for G. (cfg)
Thm 5.4 shift-reduce " " a right parser for G. (cfg)

$A \to \omega \in P$　　$A| \to \omega^R| \in P$　produce (guess)　$\omega| \to A \in P$ reduce (verify) $\to \tau(\ ) = A \to \omega \in P$

$a \in \Sigma$　　$a|a \to |$　shift (verify)　$|a \to a|$ shift (store)　$\tau(\ ) = \varepsilon$.

$\$S|x\$ \Rightarrow^{\theta_L} \$|\$$　　$S \Rightarrow^{\pi_L}_{lm} x$　　$|\theta| = |\pi| + |x|$ produce shift

$\tau(\theta_L) = \pi_L$

$\$|x\$ \Rightarrow^{\theta_R} \$S|\$$　　$S \Rightarrow^{\pi_R}_{rm} x$　　$|\theta| = |\pi| + |x|$ produce shift

$\tau(\theta_R) = (\pi_R)^R$

rightmost derivation in reversed order

$E' \to E$ augument rule
$E \to E+T \mid T*F \mid a \mid (E)$
$T \to$ — $T*F \mid a \mid (E)$
$F \to a \mid (E)$

Ex $a+a*a$, $\sqcup$　$\$|a_1+a_2*a_3\$$
$+a*a$,　$\boxed{a}$　$\Rightarrow^{a} \$a|+a*a\$$
$+a*a$　$\boxed{E} \to a$　$\Rightarrow^{E\to a} \$E|+a*a\$$
$a*a$,　$\boxed{+E} \to a$　$\Rightarrow^{+} \$E+|a*a\$$
$*a$,　$\boxed{a+E} \to a_1$　$\Rightarrow^{a} \$E+a|*a\$$
$*a$,　$\boxed{T+E} \to a_2, a_1$　$\Rightarrow^{T\to a} \$E+T|*a\$$

a,　$\boxed{*T+E} \to a_2, a_1$　$\Rightarrow^{*} \$E+T*|a\$$
ε,　$\boxed{a*T+E} \to a_2, a_1$　$\Rightarrow^{a} \$E+T*a|\$$
,　$\boxed{F*T+E} \to a_3, a_2, a_1$　$\Rightarrow^{F\to a} \$E+T*F|\$$
,　$\boxed{T+E}$ $T*F$, a_3　$\Rightarrow^{T\to T*F} \$E+T|\$$
,　$\boxed{E}$ $E+T$, a_1, $T*F$, a_2 a_3　$\Rightarrow \$E|\$ \Rightarrow^{accept} \$E'|\$$

E ⑦
E+T
①　②
T*F
① ③
a　a

Greibach normal form.

$$\left[\begin{array}{l} A \to a\alpha \in P \quad a\in\Sigma,\ \alpha \in (N\cup\Sigma)^* \\ S\to\varepsilon \end{array}\right.$$

$(N-\{S\}\cup\Sigma)^*$ → augmented rule $S' \to S \in P$ (new, old start symbol)

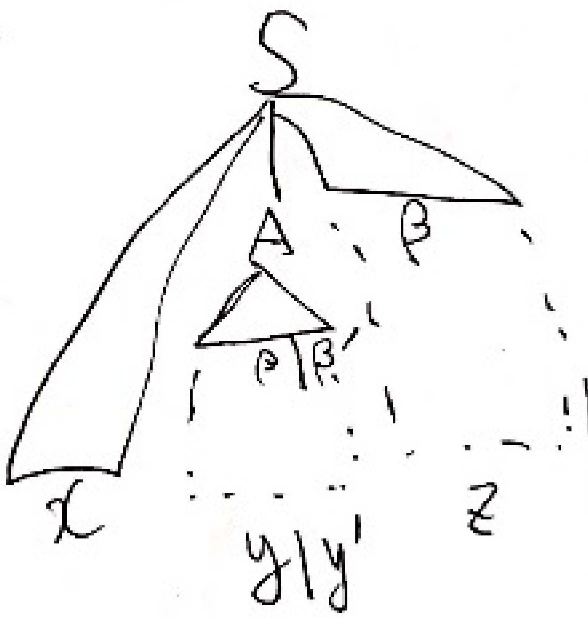
$A|\to w^R|$ $A\to\omega\in P$ — $A|a\to\alpha^R|a$ $A\to a\alpha\in P$

$a|a\to|$ $a\in\Sigma$ $\lambda|a\to|$ > one symbol lookahead left parser

A GNF → simple (s) grammar

$A\to a\alpha_1 \mid a\beta_1 \in P \Rightarrow \alpha_1=\beta_1$.

Simple grammar ... one symbol lookahead left parser is deterministic!



If $k:yz \cap k:y'z = \emptyset$, then SLL(k) parser is deterministic

$A|x\to\omega^R|x$ $A\to\omega\in P$

$a|a\to|$ $a\in\Sigma$

$\boxed{A\to\omega\in P}$

$x\in \underbrace{First_k(\omega)}_{y/y'} \oplus_k \underbrace{Follow_k(A)}_{First_k(\beta)}$

$\in\Sigma^* = (\Sigma^0\cup\Sigma^1\cup\Sigma^2\cup\ldots\cup\Sigma^k) = (\Sigma^{\le k})$

Def $First_k(a)=\{a\}$ $a\in\Sigma$

$First_k(A) = First_k(X_1)\oplus_k First_k(X_2)\oplus_k\cdots\oplus_k First_k(X_n)$

$First_k: (\Sigma\cup N)\to\Sigma^{\le k}$

⇓ Extension

$First: (N\cup\Sigma)^*\to\Sigma^{\le k}$

$First_k(\alpha)=\{k:x\in\Sigma^{\le k} \mid \alpha\Rightarrow^* x,\ x\in\Sigma^*\}$

$\alpha\in(N\cup\Sigma)^*$

Def $\oplus^{k} : \Sigma^* \times \Sigma^* \to \Sigma^{\leq k}$ (infix first_k operator)

$x \oplus_k y = k : xy.$

$x, y \in \Sigma^*$

$\text{First}_k(X_1 X_2 \cdots X_n) = \text{First}_k(X_1) \oplus_k \text{First}_k(X_2) \oplus_k \cdots \oplus_k \text{First}_k(X_n)$

Def $\text{Follow}(A) = \{ k : z \in \Sigma^{\leq k} \mid S \overset{*}{\Rightarrow} \alpha A z,\ \alpha \in (N \cup \Sigma)^*,\ z \in \Sigma^* \}$

$A \to \omega \mid \omega' \in P$ If $\text{First}_k(\omega) \oplus_k \text{Follow}_k(A) \cap \text{First}_k(\omega') \oplus_k \text{Follow}_k(A) = \emptyset$ in G

Then G is SLL(k) grammar.

$\Leftrightarrow$ SLL(k) parser is deterministic!

(구체예) $E \to E + T \mid T * F \mid a \mid (E)$
Is $T \to T * F \mid a \mid (E)$
$F \to a \mid (E)$
SLL(k)? Ans) No!

왜?. Left Recursive.

$A \to A\alpha \mid \beta$ $\quad \text{First}_k(A\alpha) \supseteq \text{First}_k(\beta)$ $(\because A \to \beta \in P)$

$\therefore \text{First}_k(A\alpha) \cap \text{First}_k(\beta) \neq \emptyset$

Removal of Left Recursion
(Left recursion $\to$ right recursion)

$A \overset{*}{\Rightarrow} \beta^* \alpha$ $\quad (A \Rightarrow A\alpha \Rightarrow AA\alpha \Rightarrow \cdots \Rightarrow A^*\alpha \Rightarrow \beta^*\alpha$

$A \to A\beta \mid \varepsilon$ (X) – left recursion

$A \to A'\alpha$
$A' \to \beta A' \mid \varepsilon$ – right recursion

$A' \overset{*}{\Rightarrow} \beta^*$

자주쓰게 Gexp 에서 left recursion을 없애면 LL(1) 일까?