

- Parse Table generation is first stage LR(0) automation or automaton
- Grammar or perspective & automation generation होती है,
- Automation द्वारा parse table generate होती है, parse table द्वारा - parsing होता है,

Different type of parser:

- LR (0) parser
- SLR (1) parser.
- LALR parser.
- CLR (1) parser.

प्रमुख अंतर difference just उनकी parse table है.

- Automation start कराते होते हैं augment होता है,
- Production body तो starting state, production head & dummy production head एवं augmentation होती है.

LR(0), SLR(1)

LR(K); K हो value 0 या 1 होता है; किसी भी स्थिति by default 0 होती है, K होती है तो आगे look ahead symbol.

$k = 0, 1 \rightarrow$ look ahead symbol.

$\frac{\text{int } a ;}{0}$ $k=0$ एवं current symbol int हो जाता है,

$\frac{\text{int } a ;}{1}$ $k=1$ एवं current symbol int हो जाता है तो
next symbol a तो आगे नहीं जाता,

augmented rule द्वारा उनका parsing start होता है,

- Sequential left shift operation होती है,

bison LALR parser
use करें,

→ ප්‍රතික rule compiler තක මැංඩි.

⇒ • එය පිටුව terminals හා non terminals නොදා තැබූව යුතු කළ ඇති.

⇒ ප්‍රතික නි completely පිටුව rule read කළ ඇති ප්‍රතික නොදා යුතු ඇති.

⇒ non terminal එක නො යුතු නියම rule නැති ඇත් මෙය open කළ - මැංඩි - start symbol.

⇒ • එය එක් ඇසුම් පිටුව shift ඇති.

⇒ • ප්‍රතික මේ යාකා අවශ්‍ය නියම already opened යුතු තැබූ ඇති.

අග්‍රාධි open කළ not necessary but • එය යාකා open කළ necessary.

$$E' \rightarrow \cdot E ; E \text{ නැත් මැංඩි}$$

$$E' \rightarrow E \cdot T F ; T \text{ නැත් මැංඩි} E \text{ නැත් මැංඩි} \text{ ලෝස්}$$

$$E' \rightarrow E T F \cdot ; F \text{ නැත් මැංඩි} E \text{ නැත්} T \text{ මැංඩි} \text{ ලෝස්.}$$

■ Given CFG ගැනී LR(0) automation කළ,

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T * F \mid F$$

$$F \rightarrow (E) \mid id$$

⇒ first step augmentation,

$$E' \rightarrow E$$

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T * F \mid F$$

$$F \rightarrow (E) \mid id$$

LR (0) Automation

