Top Down Parsing

$A \rightarrow a B c$

LL (K) — look ahead of K

- read string from left to right
- leftmost derivation

antlr → LL(*) : can decide how big a lookahead it wants at any point

$A \rightarrow A\alpha \mid \beta$ $\qquad$ $A \rightarrow \beta R$, $\quad R \rightarrow \epsilon \mid \alpha R$

$E \rightarrow E + T \mid T$

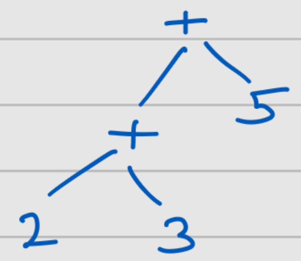
$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid id$

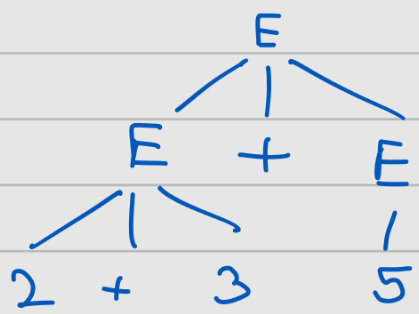
$\underbrace{E \rightarrow E + E \mid E * E \mid (E) \mid id}_{\text{this is ambiguous}}$

$2+3+5$

AST



Parse Tree



Parse Tree has non terminals. Everything is an expression

AST has only terminals. Handles operands and operators differently. (semantics)

AST — lossless representⁿ.

Program $P \longrightarrow$ AST $\xrightarrow[\text{prog}]{\text{form a}}$ Prog P' ~~Prog~~ Original Prog

$P \equiv P'$

(syntactically they remain the same)

AST is used for src to src compilation.

If we create prog from 3AC then it loses syntax. Only semantically rem same

3AC → semantically same

AST → semantically + syntactically

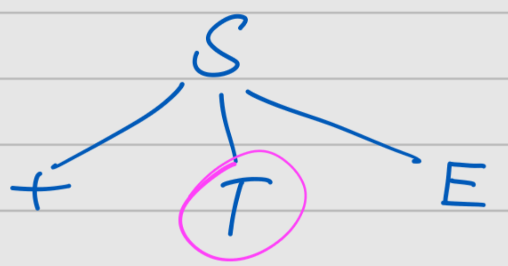
G —— L(G)

✗ LL(1) ✓ LL(1)

Grammar is not LL(1) but L(G) is an LL(1) lang.

$L(G)$ is $LL(1)$ if $\exists\, G'$ st $L(G) \equiv L(G')$

and $G' \models LL(1)$ (G' is an LL(1) grammar)



State of the parser = $\{T, a\}$

a b c d

look ahead

current non terminal which we can expand

curr look ahead symbol to produce

Cons. Parse Table

First set → if a Sentential form is nullable then $\varepsilon \in$ First set of that sent. form

$A \to BC$ $B \to \varepsilon$ $C \to c$

then $\varepsilon \notin FS(A)$ $FS(A) = \{c\}$

$A \to BC$ $B \to \varepsilon / b$ $C \to c$

$FS(B) = \{b, \varepsilon\}$ $FS(A) = \{b, c\}$

$A \to BC$ $B \to \varepsilon / b$ $C \to c / \varepsilon$

$FS(A) = \{b, c, \varepsilon\}$

$FS(B) = \{b, \varepsilon\}$

Follow Set

$\varepsilon \notin$ Follow Set of any Sent Form (chain of epsilons go to \$ eventually)

$S \to BCD\$$

follow set of B: first set of C.

if First set (C) has ε, then incl firstset of D and if it also has ε then incl \$

greek — sentential form
capital — Non terminal
small — terminal

First set can have epsilon.
First set can't have $.
Follow set : ε ✗ $ ✓