Math 322

Ch 3 — Language/Grammar ("algebra") — State machines (Graph theory)

↓ symbol manipulation

↓

Finite State Automatons (Turing Machines)

---

§3.1 Phrase-Structure Grammars

Def ① V is a vocabulary or alphabet is a non-empty set of symbols.

Ex $V = \{a, b, A, B, S, 0, 1, C\}$

② V^* the set of all possible concatenations of symbols of V.

③ a sentence or word is a finite length string of symbols

④ λ is the sentence or word of no length (null string or empty string)

⑤ Language is a subset of V^*.

Symbols: Terminals (symbols that are not meant to be replaced)
Non-Terminals (symbols that are to be replaced)

ex: article → a
article → the

Start Symbol: a Non-terminal that starts the process of replacement

ex sentence → noun ph, verb ph

Productions: rules of replacement

□ → △

G is a phrase-structure grammar.

$G = (V, T, S, P)$

V: vocabulary → $V = T \cup N$
T: set of terminals
S: start symbol
P: set of productions

P set of $z_0 \to z_1$

use it: $l\ z_0\ r \Rightarrow l\ z_1\ r$ direct derivation.

ex P_1: $aA \to aab$, P_2: $bBb \to abb$

$aAaABb \Rightarrow aAaabBb \Rightarrow aabaabBb \Rightarrow aabaabbb$
(P_1, P_1, P_2)

ex) $aAaABb \overset{*}{\Rightarrow} aabaaabb$ derivable

$L(G) = \{ w \in T^* \mid S \overset{*}{\Rightarrow} w \}$

Language of a Grammar.

---

the languages $L(G)$ are dependent on the productions $w_1 \to w_2$ with $lw_1r \Rightarrow lw_2r$

Name:	type	restriction on P
phrase-structure grammar	0	none
context-sensitive (non contracting)	1	$S \to \lambda$ is ok $lAr \to lwr$ and $w \neq \lambda$ ↑ non-term
context-free	2	$A \to$ ↑ single non-term
regular	3	$A \to aB$ term, non-term $A \to a$ term