Language and Grammar

Language: Set of strings over an alphabet

Grammar: Alphabet and rules for creating strings

Terminal symbols**:** elements of alphabet

Intermediate **(*****nonterminal*****)** ***symbols:*** *variables* enforce language syntactic restrictions

Sentence: Initial variable

Problems:

Context Free Grammar (CFG)

Grammar: Alphabet and rules for creating strings

For alphabet $\sum =$

{***a, the , John, Jill, hamburger, car drives, eats, slowly, frequently, big, juicy, brown***}

Valid strings:

John eats a hamburger

Jill drives frequently

Invalid string:

Jill the car John slowly

Problems:

Context Free Grammar (CFG)

nonterminal symbols (variables)

Possible rules for initial variable (***sentence***)

1. (*sentence*) → <noun phrase> <verb phrase>
2. (*sentence*) → <noun phrase> <verb > <direct object phrase >
3. *<noun-phrase> → <proper-noun>*
4. *→ <determiner> <adjective-list> <common-noun>*
5. *<proper-noun>* → ***John* | *Jill***
6. *<determiner>* → ***a* | *the***
7. *<common-noun>* → ***car* | *hamburger***
8. *<adjective-list> → λ | <adjective> <adjective-list>*
9. *<adjective>* → ***big* | *juicy* | *brown***
10. *<direct-object-phrase> → <noun-phrase>*
11. *<verb-phrase> → <verb> | <verb> <adverb>*
12. *<verb>* → ***drives* | *eats***
13. *<adverb>* → ***slowly* | *frequently***

recursive definition

recursive definition

Problems:

Context Free Grammar (CFG)

Derivation		Rule
(sentence)	⇒ *<noun-phrase> <verb-phrase>*	1
	⇒ *<proper-noun> <verb-phrase>*	3
	⇒ ***Jill*** *<verb-phrase>*	5
	⇒ ***Jill*** *<verb> <adverb>*	11
	⇒ ***Jill drives*** *<adverb>*	12
	⇒ ***Jill drives frequently***	13

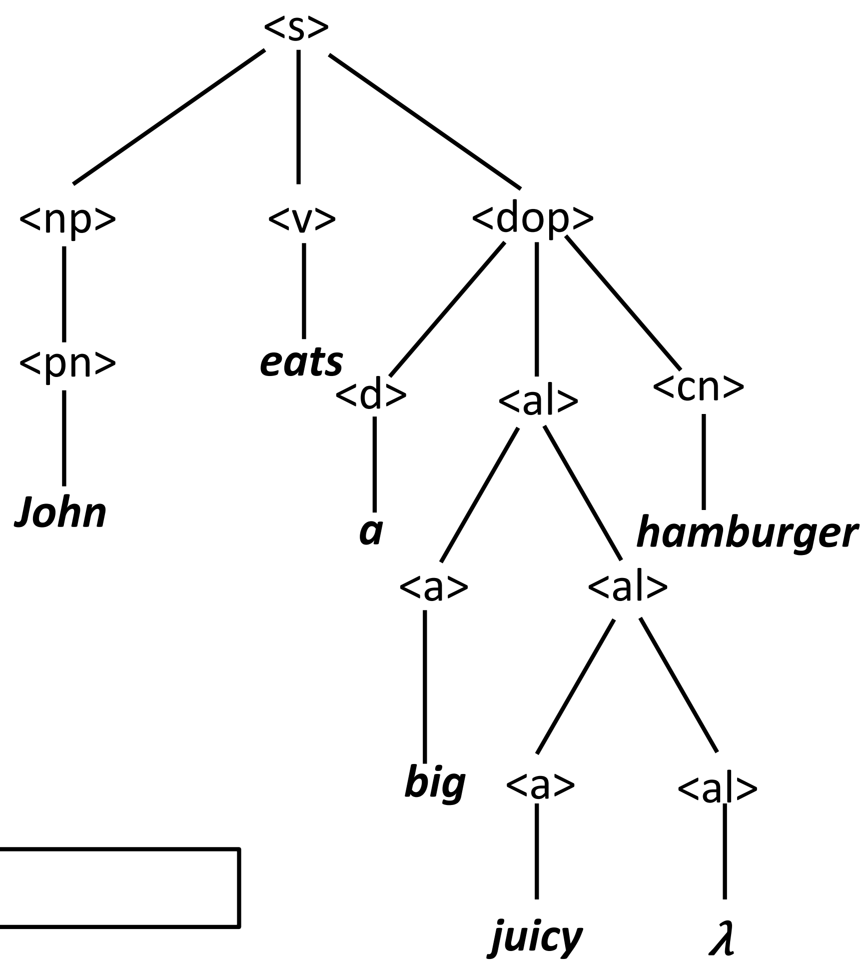
Problems:

Context Free Grammar (CFG)

Derivation	Rule(s)
(sentence) ⇒ *<noun-phrase> <verb> <direct-object-phrase>*	2
⇒ *<proper-noun> <verb> <direct-object-phrase>*	3
⇒ ***John*** *<verb> <direct-object-phrase>*	5
⇒ ***John eats*** *<direct-object-phrase>*	12
⇒ ***John eats*** *<determiner> <adjective-list><common-noun>*	10, 4
⇒ ***John eats a*** *<adjective-list><common-noun>*	6
⇒ ***John eats a*** *<adjective><adjective-list><common-noun>*	8
⇒ ***John eats a big*** *<adjective-list><common-noun>*	9
⇒ ***John eats a big*** *<adjective><adjective-list><common-noun>*	8
⇒ ***John eats a big juicy*** *<adjective-list><common-noun>*	9
⇒ ***John eats a big juicy*** *<common-noun>*	8
⇒ ***John eats a big juicy hamburger***	7

Problems:

Context Free Grammar



Problems:

Context Free Grammar

Quadruple (V, Σ, P, S) are:

V – Finite set of variables (*nonterminal* symbols)

Σ - Finite set of *terminal* symbols (alphabet)

$S \in$ V, the start symbol

P - Finite set of rules

$P \subseteq V \times (V \cup \Sigma)^*$

$A \in$ V (Variables capitalized)

$\Sigma = \{a, b, c, \ldots\}$

Problems:

$LHS \rightarrow RHS$

$A \rightarrow w$ *is the A rule*

$A \rightarrow \lambda$ *is the null rule*

$w, \lambda \in (V \cup \Sigma)^*$

$p, q, \ldots z \in (V \cup \Sigma)^*$

LHS -Variables (V)

RHS - string of terminal and nonterminal symbols - permutation of V and Σ

Context Free Grammar

Context Free Grammar:

Rule A applied wherever A occurs…

String or sentence $\boldsymbol{uAv}$

Problems:

Notation:

Definition of a rule ($\rightarrow$)

$$A \rightarrow w$$

$\boldsymbol{A}$ defined by $\boldsymbol{w}$

Derivation of a string($\Rightarrow$)

$$uAv \Rightarrow uwv$$

$\boldsymbol{uwv}$ derived from $\boldsymbol{uAv}$

$\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ are strings of terminal and nonterminal symbols - permutation of V and Σ

Context Free Grammar (CFG)

$\Sigma = \{0,1,2,3,4,5,6,7,8,9, \text{-}, (,)\}$

$P: \quad S \rightarrow (DDD)DDD\text{-}DDDD$

$\quad\quad D \rightarrow 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

D replaced by 0 or 1 or ....

allows 0 in first position

$P: \quad S \rightarrow (XDD)\ XDD\text{-}DDDD$

$\quad\quad D \rightarrow 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

$\quad\quad X \rightarrow 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

disallows 0 in first position

Problems:

Context Free Grammar (CFG)

$G = (V, \Sigma, P, S)$, $V = \{S\}$, $\Sigma = \{a\}$

$\boldsymbol{P} = \{(S, aS), (S, \lambda)\}$

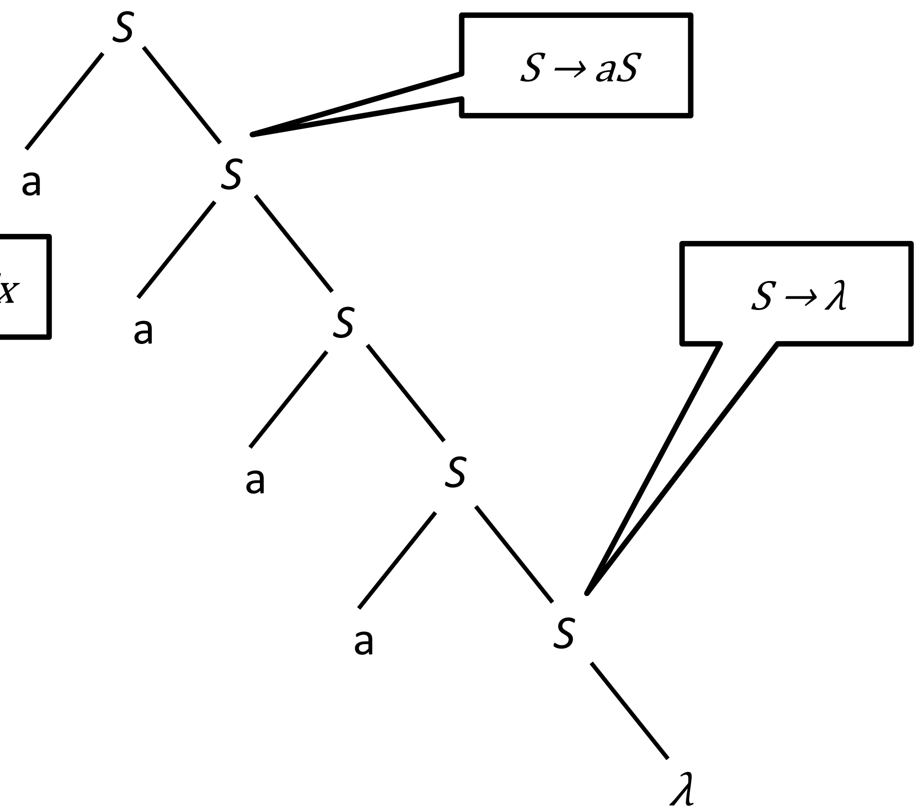
$\boldsymbol{P}$: $S \rightarrow aS$

$\quad\;\; S \rightarrow \lambda$

$\boldsymbol{P}$: $S \rightarrow aS \mid \lambda$

same P'x

$L = \{a\}^* = \{\lambda, a, aa, aaa, \ldots\}$

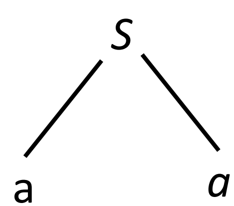


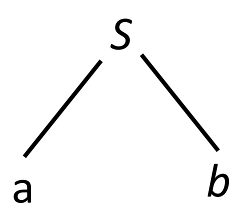
Problems:

Context Free Grammar (CFG)

$G = (V, \Sigma, P, S)$, $\Sigma = \{a, b\}$, $V = \{S\}$,

P: $S \rightarrow aa \mid ab$





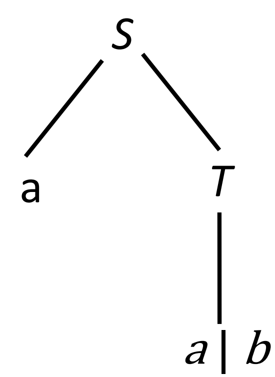
$L = \{aa, ab\}$

different grammars produce same language

$V = \{S, T\}$

$S \rightarrow aT$

$T \rightarrow a \mid b$

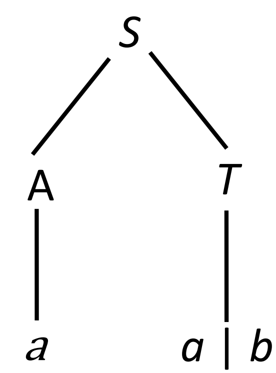


$V = \{S, A, T\}$

$S \rightarrow AT$

$A \rightarrow a$

$T \rightarrow a \mid b$



Problems:

Context Free Grammar

Definition 3.1.2: $G = (V, \Sigma, P, S)$, $v \in (V \cup \Sigma)^*$

Set of strings derivable from v

Basis: v derivable from v

$rule\ A \rightarrow \mathrm{w}$

Recursive Step:

if $u = xAy$ is derivable from v and $rule\ A \rightarrow w \in \mathrm{P}$

then xwy is derivable from v

Closure:

A string is derivable from v only if it can be generated from v with a finite number of recursive steps

v derives w in n steps: $v \Rightarrow w_n = w$, $v \overset{n}{\Rightarrow} w$

$v \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \ldots w_n = w$

Problems:

Context Free Grammar

Definition 3.1.3: $G = (V, \Sigma, P, S)$ is a context free grammar

i. A string $\boldsymbol{w} \in (V \cup \Sigma)^*$ is a ***sentential form*** of $\boldsymbol{G}$ if $\boldsymbol{S} \overset{*}{\Rightarrow} \boldsymbol{w}$ in $\boldsymbol{G}$

Sentential form includes variables and therefore not a sentence!

ii. A string $\boldsymbol{w} \in \Sigma^*$ is a ***sentence*** of $\boldsymbol{G}$ if $\boldsymbol{S} \overset{*}{\Rightarrow} \boldsymbol{w}$ in $\boldsymbol{G}$

sentence derived from $\boldsymbol{S}$

Example

(sentence) ⇒ *<noun-phrase> <verb-phrase>*

⇒ *<proper-noun> <verb-phrase>*

⇒ ***Jill*** *<verb-phrase>*

Sentential form

iii. The ***language*** of $\boldsymbol{G}$, $\boldsymbol{L(G)}$ is $\{\boldsymbol{w} \in \Sigma^* \mid \boldsymbol{S} \overset{*}{\Rightarrow} \boldsymbol{w}\}$

Language of Grammar $\boldsymbol{G}$

Problems:

Context Free Grammar (CFG)

$G = (V, \Sigma, P, S)$, $V = \{S\}$, $\Sigma = \{a\}$

P: $S \rightarrow aS$

$S \rightarrow \lambda$

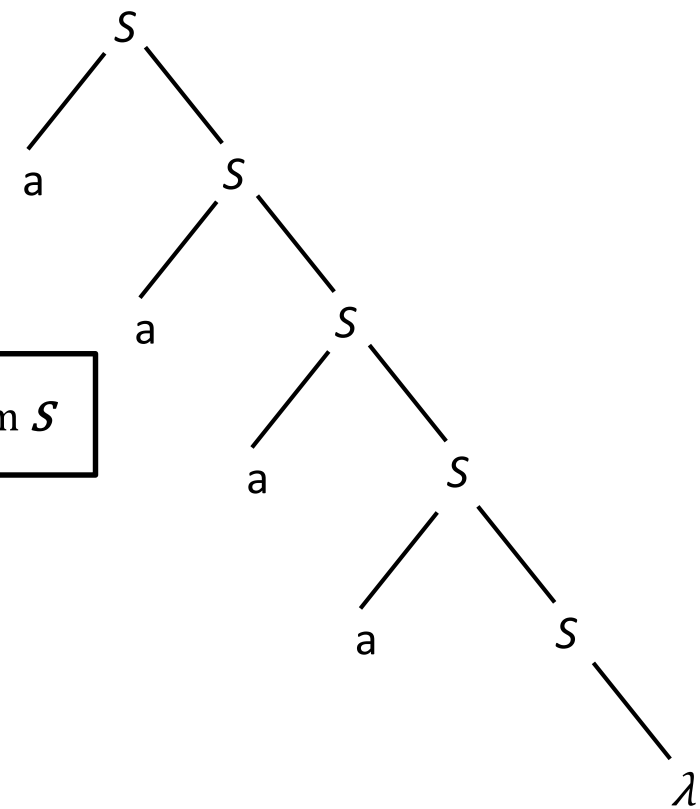
$L = \{a\}^* = \{\lambda, a, aa, aaa, \ldots\}$

$\lambda \in L(G)$? Yes, $S \Rightarrow \lambda$

derivable from ***S***

$a \in L(G)$? Yes, $S \Rightarrow aS \Rightarrow a\lambda \Rightarrow a$

$L(G) = \{a\}^* = \{a^n \mid n \geq 0\}$



Problems:

Recursive Grammars

Direct Recursive Rules:

$A \rightarrow aAb$ – recursive

$A \rightarrow aA$ – right recursive

$A \rightarrow Ab$ – left recursive

$A \rightarrow AA$ – recursive

Indirect Recursion:

$A \rightarrow B,\ B \rightarrow C,\ C \rightarrow A$

right recursive

Left recursive

$A \rightarrow aAb$	$A \rightarrow aA$	$A \rightarrow Ab$	$A \rightarrow AA$
$A \Rightarrow aAb$	$A \Rightarrow aA$	$A \Rightarrow Ab$	$A \Rightarrow AA$
$A \Rightarrow aaAbb$	$A \Rightarrow aaA$	$A \Rightarrow Abb$	$A \Rightarrow AAA$
$A \Rightarrow aaaAbbb$	$A \Rightarrow aaaA$	$A \Rightarrow Abbb$	$A \Rightarrow AAAA$
$A \Rightarrow aaaaAbbbb$	$A \Rightarrow aaaaA$	$A \Rightarrow Abbbb$	$A \Rightarrow AAAAA$

recursive

replaces versus derives

Problems:

Recursive Grammars

$G = (V, \Sigma, P, S),\ V = \{S, A\},\ \Sigma = \{a, b\}$

$P: \quad S \to AA$

$\qquad A \to AAA \mid bA \mid Ab \mid a$

leftmost derivation

rightmost derivation

alternating

$S \Rightarrow \underline{A}A$	$S \Rightarrow \underline{A}A$	$S \Rightarrow A\underline{A}$	$S \Rightarrow \underline{A}A$
$\Rightarrow a\underline{A}$	$\Rightarrow \underline{A}AAA$	$\Rightarrow \underline{A}a$	$\Rightarrow a\underline{A}$
$\Rightarrow a\underline{A}AA$	$\Rightarrow a\underline{A}AA$	$\Rightarrow AA\underline{A}a$	$\Rightarrow aAA\underline{A}$
$\Rightarrow ab\underline{A}AA$	$\Rightarrow ab\underline{A}AA$	$\Rightarrow AAb\underline{A}a$	$\Rightarrow a\underline{A}Aa$
$\Rightarrow aba\underline{A}A$	$\Rightarrow aba\underline{A}A$	$\Rightarrow A\underline{A}baa$	$\Rightarrow abA\underline{A}a$
$\Rightarrow abab\underline{A}A$	$\Rightarrow abab\underline{A}A$	$\Rightarrow Ab\underline{A}baa$	$\Rightarrow ab\underline{A}bAa$
$\Rightarrow ababa\underline{A}$	$\Rightarrow ababa\underline{A}$	$\Rightarrow \underline{A}babaa$	$\Rightarrow abab\underline{A}a$
$\Rightarrow \mathbf{ababaa}$	$\Rightarrow \mathbf{ababaa}$	$\Rightarrow \mathbf{ababaa}$	$\Rightarrow \mathbf{ababaa}$

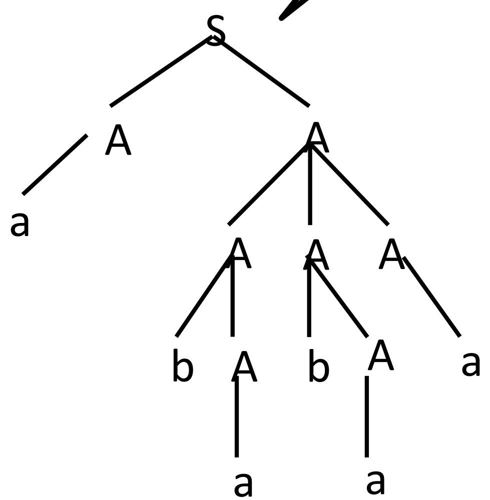
CFG
derivations produce same terminal string from different sentential forms

Problems:

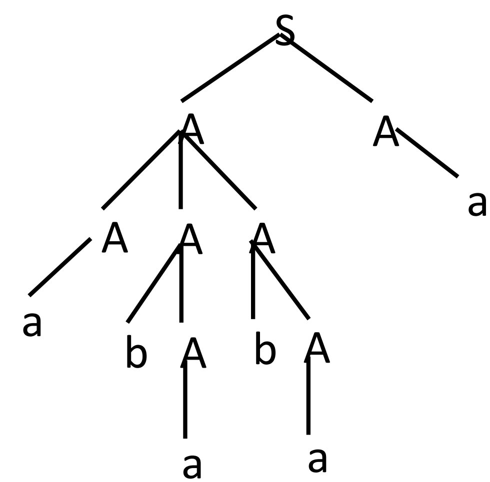
Derivation Trees

leftmost derivation

rightmost derivation



$S \Rightarrow AA$	$S \Rightarrow AA$
$\Rightarrow aA$	$\Rightarrow Aa$
$\Rightarrow aAAA$	$\Rightarrow AAAa$
$\Rightarrow abAAA$	$\Rightarrow AAbAa$
$\Rightarrow abaAA$	$\Rightarrow AAbaa$
$\Rightarrow ababAA$	$\Rightarrow AbAbaa$
$\Rightarrow ababaA$	$\Rightarrow Ababaa$
$\Rightarrow ababaa$	$\Rightarrow ababaa$



Derivation Trees

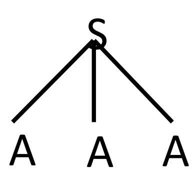
Definition 3.1.4: For $G = (V, \Sigma, P, S)$ a context free grammar and $S \stackrel{*}{\Rightarrow} w$, the **derivation tree**, DT, is built iteratively as follows:

i. Initialize DT with root S.

ii. If $A \rightarrow x_1 x_2 \ldots x_n$ with $\boldsymbol{x_i} \in (V \cup \Sigma)^*$ is the rule in the derivation applied to the string uAv then $x_1, x_2, \ldots x_n$ are children of A

iii. If $A \rightarrow \lambda$ is the rule in the derivation applied to the string uAv then add λ as the only child of A

$S \rightarrow AAA$



Problems:

Examples of Grammars

Example 3.2.1

$G = (V, \Sigma, P, S)$, $V = \{S, B\}$, $\Sigma = \{a, b\}$

P: $S \rightarrow aSa \mid aBa$

$B \rightarrow bB \mid b$

Rule $S \rightarrow aSa$	n- 1 times
Rule $S \rightarrow aBa$	1 time
Rule $B \rightarrow bB$	m-1 times
Rule $B \rightarrow b$	1 time

$S \Rightarrow aSa$
$\Rightarrow aaSaa$
$\Rightarrow aaaSaaa$
$\Rightarrow aaaaBaaaa$
$\Rightarrow aaaabBaaaa$
$\Rightarrow aaaabbBaaaa$
$\Rightarrow aaaabbbaaaa$

$L(G) = \{ a^n b^m a^n \mid n > 0, m > 0 \}$

Problems:

Examples of Grammars

Example 3.2.2

$G = (V, \sum, P, S)$, $V = \{S, A\}$, $\sum = \{a, b, c, d\}$

P: $S \rightarrow aSdd \mid A$

$A \rightarrow bAc \mid bc$

Rule $S \rightarrow aSdd$	n times
Rule $S \rightarrow A$	1 time
Rule $A \rightarrow bAc$	m-1 times
Rule $A \rightarrow bc$	1 time

$S \Rightarrow aSdd$
$\Rightarrow aaSdddd$
$\Rightarrow aaaSdddddd$
$\Rightarrow aaaAdddddd$
$\Rightarrow aaabAcdddddd$
$\Rightarrow aaabbccdddddd$

$L(G) = \{ a^n b^m c^m d^{2n} \mid n \geq 0, m > 0 \}$

Problems:

Examples of Grammars

Example 3.2.4

$G = (V, \Sigma, P, S)$, $V = \{S\}$, $\Sigma = \{a, b\}$

P: $S \to aSb \mid aSbb \mid \lambda$

$\boldsymbol{n}$ = number $\boldsymbol{a}$'s
$\boldsymbol{m}$ = number $\boldsymbol{b}$'s
Rule $S \to aSb$ produces $m = n$
Rule $S \to aSbb$ produces $m = 2n$
combination of rules $0 \le n \le m \le 2n$

$S \Rightarrow aSb$	**(n, m)**
$\Rightarrow aaSbb$	(2, 2)
$\Rightarrow aaaSbbb$	(3, 3)
$\Rightarrow aaaaSbbbbb$	(4, 5)
$\Rightarrow aaaaaSbbbbbbb$	(5, 7)
$\Rightarrow aaaaaaSbbbbbbbbb$	(6, 9)
$\Rightarrow aaaaaaaSbbbbbbbbbbb$	(7, 11)
$\Rightarrow aaaaaaa\lambda bbbbbbbbbbb$	
$\Rightarrow aaaaaaabbbbbbbbbbb$	

$L(G) = \{a^n b^m \mid 0 \le n \le m \le 2n\}$

Problems:

Examples of Grammars

Example 3.2.5

$G = (V, \Sigma, P, S)$, $V = \{S, B\}$, $\Sigma = \{a, b, c\}$

P: $S \rightarrow abScB \mid \lambda$

$B \rightarrow bB \mid b$

Rule $S \rightarrow abScB$ n times
$S \rightarrow \lambda$

$L(G) = \{(ab)^n(cb^{m_n})^n \mid n \geq 0,\, m_n \geq 0\}$????

$L(G) = \{\lambda \cup \{(ab)^n\, (cb^{m_1}) \ldots (cb^{m_i}) \ldots (cb^{m_n}) \mid$
$n \geq 1,\, m_i \geq 1,\, i = 1, 2, \ldots n\}\}$

$L(G) = \{(ab)^n(cb^{m_i})^n \mid n \geq 0,\, m_i \geq 1,\, i = 0, 1, 2, \ldots n\}\}$

$S \Rightarrow abScB$
$\Rightarrow ababScBcB$
$\Rightarrow abab\lambda cBcB$
$\Rightarrow (ab)^2(cBcB)$
$\Rightarrow (ab)^2(cbB)(cB)$
$\Rightarrow (ab)^2(cbbB)(cB)$
$\Rightarrow (ab)^2(cbbbB)(cB)$
$\Rightarrow (ab)^2(cbbbb)(cB)$
$\Rightarrow (ab)^2(cbbbb)(cbB)$
$\Rightarrow (ab)^2(cbbbb)(cbb)$
$\Rightarrow (ab)^2(cb^4)(cb^2)$

Example of Equivalent Grammars

Example 3.2.6

$G_1 = (V, \Sigma, P, S)$, $V = \{S, A, B\}$, $\Sigma = \{a, b\}$

P: $S \to AB$

$A \to aA \mid a$

$B \to bB \mid \lambda$

Rule $S \to AB$	1 time
Rule $A \to aA \mid a$	$>$ 0 times
Rule $B \to bB \mid \lambda$	$\geq$ 0 times

$L(G) = \{a^+ b^*\}$

$L(G)$ is a left to right grammar

$G_2 = (V, \Sigma, P, S)$, $V = \{S, A\}$, $\Sigma = \{a, b\}$

P: $S \to aS \mid aA$

$A \to bA \mid \lambda$

$L(G) = \{a^+ b^*\}$

Problems:

Example of Equivalent Grammars

Example 3.2.7 – Exactly 2 *b's*

$G_1 = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \rightarrow AbAbA$

$A \rightarrow aA \mid \lambda$

Problems:

$L(G) = \{a^* ba^* ba^*\}$

$G_2 = (V, \Sigma, P, S),\ V = \{S, A, C\}, \Sigma = \{a, b\}$

P: $S \rightarrow aS \mid bA$

$A \rightarrow aA \mid bC$

$C \rightarrow aC \mid \lambda$

produces first ***a****

produces 2nd ***a****

$L(G) = \{a^* ba^* ba^*\}$

Examples of Grammars

Example 3.2.8 – at least 2 *b's*

$G_1 = (V, \Sigma, P, S)$, $V = \{S, A\}$, $\Sigma = \{a, b\}$

P: $S \rightarrow AbAbA$

*separated by **b**'s*

$A \rightarrow aA \mid bA \mid \lambda$

produces $(a \cup b)^*$

$L(G) = \{(a \cup b)^* b (a \cup b)^* b (a \cup b)^*\}$

$G_2 = (V, \Sigma, P, S)$, $V = \{S, A, C\}$, $\Sigma = \{a, b\}$

P: $S \rightarrow aS \mid bA$

$A \rightarrow aA \mid bC$

a^*b

$S \rightarrow bA$ and $A \rightarrow bC$
produce 1st and 2nd b

$C \rightarrow aC \mid bC \mid \lambda$

$(a \cup b)^*$

$L(G) = \{a^* b a^* b (a \cup b)^*\}$

Problems:

Examples of Grammars

Example 3.2.9

$G = (V, \Sigma, P, S),\ V = \{S, O\}, \Sigma = \{a, b\}$

P: $S \rightarrow aO \mid bO \mid \lambda$

$O \rightarrow aS \mid bS$

$S\ (O)$ in the sentential string indicates even (odd) number of terminal symbols

$L\{G\} = \{\ w \in \Sigma^* \mid w$ *is* an even length string$\}$

$S \Rightarrow aO$
$\Rightarrow aaS$
$\Rightarrow aabO$
$\Rightarrow aabbS$
$\Rightarrow aabb\lambda$
$\Rightarrow aabb$

$S \Rightarrow aO$
$\Rightarrow aaS$
$\Rightarrow aaaO$
$\Rightarrow aaaaS$
$\Rightarrow aaaa\lambda$
$\Rightarrow aaaa$

Problems:

Examples of Grammars

Example 3.2.10

$G = (V, \Sigma, P, S)$, $V = \{S, B, C\}$, $\Sigma = \{a, b\}$

P: $S \to aS \mid bB \mid \lambda$

$B \to aB \mid bS \mid bC$

$C \to aC \mid \lambda$

Produces terminal strings with ***even number*** of *b 's*

and ***even or odd number*** of *a's*

paths to λ:

$S \to bB$ (odd number of *b's*) $S \to bB$ (odd number of *b's*)

$B \to bS$ (even number of *b's*) $B \to bC$ (even number of *b's*)

$S \Rightarrow aS$
$\Rightarrow abB$
$\Rightarrow abaB$
$\Rightarrow ababS$
$\Rightarrow ababbB$
$\Rightarrow ababbbC$
$\Rightarrow ababbb\lambda$
$\Rightarrow ababbb$

Problems:

Examples of Grammars

Example 3.2.11

$G = (V, \Sigma, P, S)$, $V = \{S, A, B, C\}$, $\Sigma = \{a, b\}$

P: $S \to aB \mid bA \mid \lambda$

$A \to aC \mid bS$

$B \to aS \mid bC$

$C \to aA \mid bB$

only 1 variable
In string

Variable in String	Interpretation
S	Even number a's and b's
A	Even number a's and odd number b's
B	Odd number a's and even number b's
C	Odd number a's and b's

Indirect recursion

$S \to B \mid A$

$A \to C \mid S$

$B \to S \mid C$

$C \to A \mid B$

$L\{G\} = \{\, w \in \Sigma^* \mid w$ *is* an even number *a*'s and *b*'s$\}$
(string terminates from S)

Problems:

Examples of Grammars

Example 3.2.12

$G = (V, \Sigma, P, S)$, $V = \{S, B, C\}$, $\Sigma = \{a, b, c\}$

P: $S \to bS \mid cS \mid aB \mid \lambda$

$B \to aB \mid cS \mid bC \mid \lambda$

$C \to aB \mid bS \mid \lambda$

$S \Rightarrow aB$
$\Rightarrow abC$

1. cannot produce string *abc*
2. B occurs when previous terminal is a $S \to aB$, $B \to aB$, $C \to aB$)
3. C present only when preceded by ab $S \to aB \mid B \to aB$ then $\boldsymbol{B \to bC}$
4. C rules cannot produce terminal c $C \to aB \mid bS \mid \lambda$

***a** always precedes B*
only B is replaced by C

Problems:

Examples of Languages

$G = (V, \Sigma, P, S)$, $V = \{S, T\}$, $\Sigma = \{0, 1\}$

P: $S \rightarrow 1T \mid 0T0T0TS \mid \lambda$

$T \rightarrow 1T \mid \lambda$

$L(G) = \{w \in \Sigma^* \mid$ number 0's in w is a multiple of 3 $\}$

$G = (V, \Sigma, P, S)$, $V = \{S, T, Y\}$, $\Sigma = \{0, 1\}$

P: $S \rightarrow 00T \mid 1Y \mid \lambda$

$T \rightarrow 00T \mid \lambda$

$Y \rightarrow 11Y \mid \lambda$

Recursive and string terminates with λ

$L(G) = \{w \in \Sigma^* \mid w$ consists of only 0's and $|w|$ is even $\cup$

w consists of only 1's and $|w|$ is odd$\}$

Problems:

Examples of Languages

$G = (V, \Sigma, P, S)$, $V = \{S, A, B, C\}$, $\Sigma = \{a, b, c\}$

P: $S \rightarrow aA \mid bB \mid cC$

$A \rightarrow aA \mid \lambda$

$B \rightarrow bB \mid \lambda$

$C \rightarrow cC \mid \lambda$

$L(G) = \{w \in \Sigma^* \mid w$ has ***only one*** terminal symbol$\}$

$L(G) = \{a^n \cup b^n \cup c^n \mid n \geq 1\}$

Problems:

Examples of Languages

$G = (V, \Sigma, P, S)$, $V = \{S, A, B, C\}$, $\Sigma = \{a, b, c\}$

$$
\begin{aligned}
P: \quad & S \rightarrow ABC \\
& A \rightarrow aA \mid \lambda \\
& B \rightarrow bB \mid \lambda \\
& C \rightarrow cC \mid \lambda
\end{aligned}
$$

$$L(G) = \{a^i b^j c^k \mid i, j, k \geq 0\}$$

$$
\begin{aligned}
P: \quad & S \rightarrow ABC \\
& A \rightarrow aA \mid a \\
& B \rightarrow bB \mid b \\
& C \rightarrow cC \mid c
\end{aligned}
$$

$$L(G) = \{a^i b^j c^k \mid i, j, k \geq 1\}$$

Problems:

Examples of Languages

$G = (V, \Sigma, P, S)$, $V = \{S, X\}$, $\Sigma = \{a, b, c\}$

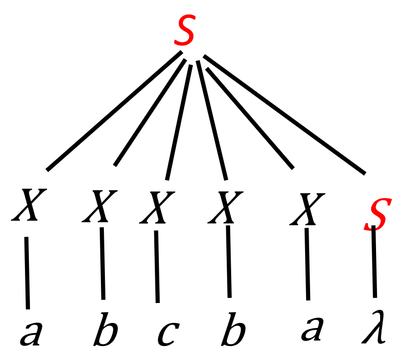
P: $\quad S \rightarrow XXXXXS \mid \lambda$

$\quad\quad X \rightarrow a \mid b \mid c$

Recursive and terminates string with λ

$L(G) = \{w \in \Sigma^* \mid length(w) \text{ is a multiple of } 5\}$

String $abcba \in L(G)$



Problems:

Examples of Languages

Recursive Langauge From Chapter 1 and 2:

$\sum = \{a, b\}$

Basis: $\lambda \in L$

Recursive Step: if $\boldsymbol{w} \in L$ then $\boldsymbol{awb} \in L$

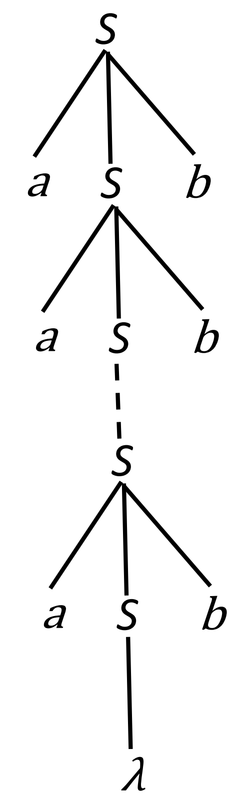
$L(G) = \{\lambda, a\lambda b, aabb, aaabbb, \ldots\} = \{a^n b^n \mid n \geq 0\}$

From Chapter 3:

$G = (V, \sum, P, S), V = \{S\}, \sum = \{a, b\}$

P: $\quad S \rightarrow aSb \mid \lambda$

$L(G) = \{w \in \sum^* \mid \mathrm{w} = a^n b^n \mid n \geq 0\}$



Problems:

Examples of Languages

Problem 3.9

$G = (V, \Sigma, P, S),\ V = \{S, B, C\}, \Sigma = \{a, b, c\}$

$L(G) = \{a^n b^m c^i \mid 0 \le n + m \le i\}$

P: $S \to aSc \mid B \mid \lambda$

$B \to bBc \mid C$

$C \to cC \mid \lambda$

Problems:

perform $S \to aSc$ *n times then*

perform $B \to bBc$ *m times* then (or)

perform $B \to C$

perform $C \to cC$ $(i - n - m)$ *incremental times*

$L(G) = \{\lambda\} \cup \{c^i \mid n, m = 0, i \ge 1\} \cup \{a^n c^i \mid m = 0, n \le i\} \cup$

$\{b^m c^i \mid n = 0, m \le i\} \cup \{a^n b^m c^i \mid n \ge 1, m \ge 1, 0 \le n + m \le i\}$

$n, m, i = 0$

Examples of Languages

Problem 3.10

$G = (V, \Sigma, P, S)$, $V = \{S\}$, $\Sigma = \{a, b\}$

$L(G) = \{a^m b^n \mid 0 \leq n \leq m \leq 3n\}$

$L(G) =$ {number *a*'s at least number of *b*'s and

no more than 3 times number *b*'s}

P: $S \rightarrow aSb \mid aaSb \mid aaaSb \mid \lambda$

0 ***a's*** and 0 ***b's***

one ***a*** and one ***b***

two ***a***'s and one ***b***

three ***a***'s and one ***b***

$L(G) = \{\lambda, ab, a^2b, a^3b, a^2b^2, a^3b^2, a^4b^2, a^5b^2, a^6b^2, \ldots\}$

Problems:

Examples of Languages

Problem 3.11

$G = (V, \Sigma, P, S)$, $V = \{S, T, Y\}$, $\Sigma = \{a, b\}$

$L(G) = \{a^m b^i a^n \mid i = m + n\}$

$L(G) = \{a^m b^{m+n} a^n \mid m + n \geq 0\}$

$L(G) = \{a^m b^m b^n a^n \mid m + n \geq 0\}$

$L(G) = \{a^m b^m \mid m \geq 0\} \{b^n a^n \mid n \geq 0\}$

Concatenation of two sets

P: $S \rightarrow TY$

$T \rightarrow aTb \mid \lambda$ *m times*

$Y \rightarrow bYa \mid \lambda$ *n times*

Problems:

Regular Grammars

Definition 3.3.1

Regular Grammar, G, is a ***restrictive*** context-free grammar

$G = (V, \Sigma, P, S)$, $V = \{S, A, B\}$ and $\Sigma = \{a\}$ with rules in the form:

$A \to a \mid aB \mid \lambda$

$A \to \lambda \mid \Sigma V \mid \Sigma$

Formal rules definition: $P \subseteq V \, x \, \{\lambda \cup \Sigma V \cup \Sigma\}$

Only one variable and variable is right most symbol in the string

Example

$G = (V, \Sigma, P, S)$, $V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \to aS \mid aA$

$A \to bA \mid \lambda$

$S \Rightarrow aS$
$\Rightarrow aS$
$\Rightarrow aaS$
$\Rightarrow aabA$
$\Rightarrow aabbA$
$\Rightarrow aabb\lambda$
$\Rightarrow aabb$

Problems:

Regular Grammars

Example 3.3.1

Context Free Grammar (CFG)

$G_1 = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a, b\}$

P: $\quad S \rightarrow abSA \mid \lambda$

$\quad\quad A \rightarrow Aa \mid \lambda$

$L(G) = \lambda \cup (ab)^+a^*$

Regular CFG

$G_2 = (V, \Sigma, P, S),\ V = \{S, A, B\}, \Sigma = \{a, b\}$

P: $\quad S \rightarrow aB \mid \lambda$

$\quad\quad B \rightarrow bS \mid bA$

$\quad\quad A \rightarrow aA \mid \lambda$

$L(G) = \lambda \cup (ab)^+a^*$

$S \Rightarrow abSA$	$S \Rightarrow aB$
$\Rightarrow abSA$	$\Rightarrow aA$
$\Rightarrow ababSAA$	$\Rightarrow abA$
$\Rightarrow abab\lambda AA$	$\Rightarrow abaA$
$\Rightarrow abab\lambda AaA$	$\Rightarrow ababA$
$\Rightarrow ababAaAa$	$\Rightarrow ababaA$
$\Rightarrow abab\lambda aAa$	$\Rightarrow ababaaA$
$\Rightarrow ababaAa$	$\Rightarrow ababaa\lambda$
$\Rightarrow ababa\lambda a$	$\Rightarrow \mathbf{ababaa}$
$\Rightarrow \mathbf{ababaa}$	

Languages and Grammars

Grammar: Variables, alphabet and rules for creating strings

$P(\Sigma^*)$ is the set of all languages

A language is a subset of Σ^*

Not every language derived from a ***CFG***

Not every language derived from a ***Regular Grammar***

Regular Grammar $\subset$ ***CFG***

Regular Grammar is generated by a ***Regular Expression***

Regular Language described by a ***Regular Grammar***

Regular Language $\subset$ ***CFL***

Problems:

Leftmost Derivations

Theorem 3.5.1

G = (V, ∑, P, S), **string w** $\in L(G)$

iff there is a leftmost derivation of w from S $S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \ldots w_n = w$

Independent rule applications to build leftmost derivation of w

i. Find first w_k such that sentential form is not a leftmost derivation (no k indicates leftmost derivation)

ii. Reorder $k + 1$ rule application as leftmost derivation.

iii. Repeat i and ii up to $n - k$ times as necessary

leftmost derivations of terminal strings are assured

NO assurance of derivations for all sentential forms

Leftmost Derivations

Theorem 3.5.1

G = (V, ∑, P, S), **string w** $\in L(G)$

iff there is a leftmost derivation of w from S $S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \ldots w_n = w$

Independent rule applications to build leftmost derivation of w

$S \Rightarrow abAV \Rightarrow abAc \Rightarrow abbc$

$V \rightarrow c$

$A \rightarrow b$

$S \Rightarrow abAV \Rightarrow abbV \Rightarrow abbc$

$A \rightarrow b$

$V \rightarrow c$

reordered rules ensure left – right derivation

leftmost derivations of terminal strings are assured

NO assurance of derivations for all sentential forms

Leftmost Derivations

leftmost derivations of all sentential strings are NOT assured

$G = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \rightarrow AB$

$A \rightarrow aA \mid \lambda$

$B \rightarrow bB \mid \lambda$

$L(G) = a^*b^*$

leftmost derivation

rightmost derivation

$S \Rightarrow AB$	$S \Rightarrow AB$
$\Rightarrow \lambda B$	$\Rightarrow A\lambda$
$\Rightarrow B$	$\Rightarrow A$

no leftmost derivation with *A in sentential form*

Problems:

Leftmost Derivations and Ambiguity

Definition 3.5.2 G is ***ambiguous*** if w $\in L(G)$ can be derived by two leftmost derivations (different sentential strings)

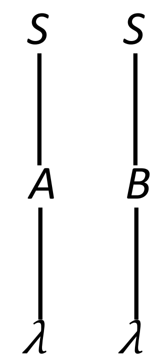
Ambiguity is property of grammar not the language

$$G = (V, \Sigma, P, S),\ V = \{S, A, B\},\ \Sigma = \{a, b\}$$

$$P: \quad S \rightarrow A \mid B$$

$$A \rightarrow aA \mid \lambda$$

$$B \rightarrow bB \mid \lambda$$



ambiguous grammars when strings have ≥2 distinct derivations

Leftmost Derivations and Ambiguity

ambiguous grammars have ≥2 distinct leftmost derivations of a terminal string derived from different sentential strings

$G = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a\}$

P: $S \rightarrow aS \mid Sa \mid a$

$L(G) = a^+$

ambiguous

$S \Rightarrow aS$	$S \Rightarrow Sa$
$\Rightarrow aaS$	$\Rightarrow Saa$
$\Rightarrow aaa$	$\Rightarrow aaa$
$\Rightarrow aaa$	$\Rightarrow aaa$

$G = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a\}$

P: $S \rightarrow aS \mid a$

$L(G) = a^+$

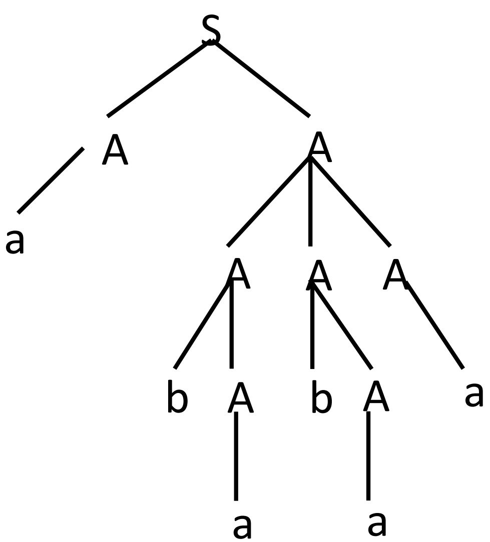
unambiguous

Problems:

Leftmost Derivations and Ambiguity

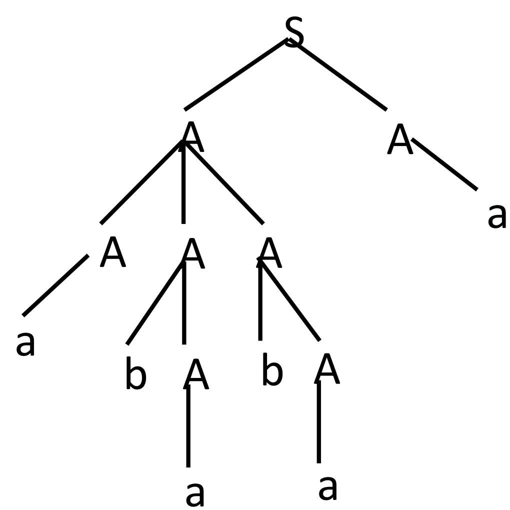
$$S \rightarrow AA \mid \lambda$$

$$A \rightarrow AAA \mid bA \mid Ab \mid a$$



$S \Rightarrow AA$
$\Rightarrow aA$
$\Rightarrow aAAA$
$\Rightarrow abAAA$
$\Rightarrow abaAA$
$\Rightarrow ababAA$
$\Rightarrow ababaA$
$\Rightarrow ababaa$

produces same terminal string from different sentential forms



$S \Rightarrow AA$
$\Rightarrow AAAA$
$\Rightarrow aAAA$
$\Rightarrow abAAA$
$\Rightarrow abaAA$
$\Rightarrow ababAA$
$\Rightarrow ababaA$
$\Rightarrow ababaa$

Leftmost Derivations and Ambiguity

Example 3.5.2

$G = (V, \Sigma, P, S),\ V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \to bS \mid Sb \mid a$

$L(G) = b^*ab^*$

$S \Rightarrow bS$	$S \Rightarrow Sb$
$\Rightarrow bSb$	$\Rightarrow bSb$
$\Rightarrow bab$	$\Rightarrow bab$

ambiguous grammar – ability to generate b* in either direction

$S \Rightarrow bS$	$S \Rightarrow Sb$
$\Rightarrow bS$	$\Rightarrow Sb$
$\Rightarrow bbS$	$\Rightarrow Sbb$
$\Rightarrow bbSb$	$\Rightarrow bSbb$

Problems:

Leftmost Derivations and Ambiguity

Example 3.5.2 (*continued*)

eliminates ability to generate b in either direction (**unambiguous**)*

$G_1 = (V, \Sigma, P, S)$,

$V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \to bS \mid aA$

$A \to bA \mid \lambda$

$S \overset{n}{\Rightarrow} b^nS$

$\Rightarrow b^naA$

$\overset{m}{\Rightarrow} b^nab^mA$

$\Rightarrow b^nab^m \lambda$

$\Rightarrow b^nab^m$

b^n then then b^m

$G_2 = (V, \Sigma, P, S)$,

$V = \{S, A\}, \Sigma = \{a, b\}$

P: $S \to bS \mid A$

$A \to Ab \mid a$

$S \overset{n}{\Rightarrow} b^nS$

$\Rightarrow b^nA$

$\overset{m}{\Rightarrow} b^nAb^m$

$\Rightarrow b^nab^m$

b^n then then b^m

$L(G) = b^*ab^*$