4. FORMAL LANGUAGES

CHAPTER - 4
FORMAL LANGUAGES

In this chapter we introduce the concepts of grammars and formal languages and discuss the chomsky classification of languages.

4.1 BASIC DEFINITIONS AND EXAMPLES

The theory of formal languages is used in the field of *Linguistics*- to define valid sentences and give structural descriptions of sentences. Linguists wanted to define a formal grammar (to describe the rules of grammar in a mathematical way) to describe English, would make language translation using computers easy.

Two types of sentences(S) in English to formalise the construction of sentences are listed below in the rules.

Rules or Productions(P)

P1 : S→<noun> <verb> <adverb>

S→<noun><verb>

<noun>→ Ram

<noun> → San

<verb>→ ran

<verb> → ate

<adverb> → slowly

<adverb> → quickly

Note :

S – variable to denote a sentence

→ – represents a rule meaning that the word on the right side of the arrow can replace the word on the left side of the arrow.

P – collection of rules (or) productions.

The sentences are derived from the above mentioned productions by:

(i) starting with S

(ii) replacing words using the productions

(iii) terminating when a string of terminals is obtained.

Example 4.1

(i) $S \rightarrow$<noun><verb><adverb>

$S \rightarrow$ Ram ate slowly

(ii) $S \rightarrow$ <noun><verb>

$S \rightarrow$ Sam ran

With this background we can give the definition of a grammar given by *'Noam Chomsky'*.

4.1.1 Definition of a grammar

A *phrase-structure grammar* (or) simply a *grammar* is (V_N, Σ, P, S), where

(i) V_N is a finite nonempty set whose elements are called *variables* or *non terminals*.

(ii) Σ is a finite nonempty set whose elements are called *terminals*.

(iii) $V_N \cap \Sigma = \phi$

(iv) S is a special variable (i.e. an element of V_N) called the *start symbol*.

(v) P is a finite set whose elements are $\alpha \rightarrow \beta$, whose α and β are strings on $V_N \cup \Sigma$. α has atleast one symbol from V_N. Elements of P are called *productions* or *production rules* or *rewriting rules*.

Note :

The set of productions is the *Kernal* of grammars and language specification. We observe the following regarding the production rules.

(i) Reverse substitution is not permitted. For example, if $S \rightarrow AB$ is a production then we can replace S by AB, but we cannot replace AB by S.

(ii) No inversion operation is permitted. For example, if $S \rightarrow AB$ is a production, it is not necessary that $AB \rightarrow S$ is a production.

Example 4.2

$G = (V_N, \Sigma, P, S)$ is a grammar where V_N, Σ, P, S from P1 is (defined already):

V_N = {<sentence>,<noun>,<verb>,<adverb>}

Σ = {Ram, Sam, ate, ran}

S = <sentence>

P = P1

4.1.2 Formal definition of derivation

If $\alpha \to \beta$ is a production in a grammar G and γ, δ are any two strings on $V_N \cup \Sigma$, then we say $\gamma\alpha\delta$ directly derives $\gamma\beta\delta$ in G. ((i.e.) $\gamma\alpha\delta \underset{G}{\Rightarrow} \gamma\beta\delta$). This process is called *one-step derivation*.

In particular, if $\alpha \to \beta$ is a production, then $\alpha \underset{G}{\Rightarrow} \beta$.

Example 4.3

G = ({S}, {0,1}, {S→0S1, S→01}, S) then the derivation is:

$S \underset{G}{\Rightarrow} 0S1$

$\therefore$ $0S1 \underset{G}{\Rightarrow} 0011$ is a one step derivation, where S is replaced by 01.

Note :

The notation $\overset{*}{\underset{G}{\Rightarrow}}$, represents a relation R on $(V_N \cup \Sigma)^*$

i.e. (i) $\alpha R \beta$ if $\alpha \underset{G}{\Rightarrow} \beta$ (production rule is applied once),

(ii) $\alpha R \beta$ if $\alpha \overset{*}{\underset{G}{\Rightarrow}} \beta$ (*indicates production rule is applied more than once)

4.1.3 Formal definition of languages

The *language* generated by a grammar G (denoted by L(G)) is defined as $\{w \in \Sigma^* \mid S \overset{*}{\underset{G}{\Rightarrow}} w\}$.

The elements of L(G) are called *sentences*.

Stated in simple way, L(G) is the set of all terminal strings derived from the start symbol S.

Note :

(i) G_1 and G_2 are equivalent if $L(G_1) = L(G_2)$

(ii) $A \to \alpha_1$, $A \to \alpha_2$....... $A \to \alpha_m$ are A-productions, rewritten as $A \to \alpha$, $|\alpha_2|$.....α_m

Examples 4.4

If G=({S}, {a}, {S → SS}, S), find the language generated by G.

Solution :

$L(G) = \phi$, since the only production $S \to SS$ in G has no terminal on the right hand side.

Examples 4.5

Let $G = (\{S,C\}, \{a,b\}, P,S\}$, where P consists of $S \to aCa$, $C \to aCa \mid b$. Find L(G)

Solution:

(i) $S \Rightarrow a\,Ca$

$\Rightarrow a\,b\,a$

$\therefore\ aba \in L(G)$

(ii) $S \Rightarrow aCa$ (from the given production $S \to aCa$)

$\Rightarrow aaCa\,a$ ($C \to aCa$)

$\Rightarrow a^n Ca^n$ (applied $C \to aCa$ production for $(n-1)$ times)

$\Rightarrow a^n ba^n$ ($C \to b$)

Two terminal strings which are derived from the above example are aba, $a^n ba^n$, where $aba \subseteq a^n ba^n$

$L(G) \subseteq \{a^n ba^n \mid n \geq 1\}$

Hence $L(G) = \{a^n ba^n \mid n \geq 1\}$

Examples 4.6

Let $G = (\{S,A_1,A_2\}, \{a,b\}, P,S)$, where P consists of

(i) $S \to aA_1A_2a$

(ii) $A_1 \to baA_1A_2b$

(iii) $A_2 \to A_1ab$

(iv) $aA_1 \to baa$

(v) $bA_2b \to abab$.

Test whether $w = baabbabaaabbaba$ is in L(G).

Solution :

(i) Start with S-production

(ii) Apply suitable production to derive w.

(iii) Substring which is replaced by the production is underlined.

$$
\begin{aligned}
S &\Rightarrow \underline{aA_1}A_2a \\
&\Rightarrow baa\underline{A_2}a \text{ (by production (iv))} \\
&\Rightarrow ba\underline{aA_1}aba \text{ (by production (iii))} \\
&\Rightarrow baab\underline{aA_1}A_2baba \text{ (by production (ii)} \\
&\Rightarrow baabbaa\underline{A_2}baba \text{ (by production (ii)} \\
&\Rightarrow baabba\underline{aA_1}abbaba \text{ (by production (iii)} \\
&\Rightarrow baabba\underline{baa}abbaba = w \text{ (by production (iv))}
\end{aligned}
$$

Therefore, the given $w \in L(G)$

Examples 4.7

Construct a grammar generating $L = \{0^n1^{2n} \mid n \geq 1\}$

Solution :

G=({S},{0,1}, P,S)

Productions (P):

$S \rightarrow 0S11$

$\underset{G}{\Rightarrow}$

$S \rightarrow 011$

Note :

(i) starts with 0 (odd or even) number of 0'S

(ii) ends with 11 (even number of 1's (i.e.) 2n)

Examples 4.8

If G=({S}, {0,1}, {S → 0S1, S → λ},S), find L(G)

Solution :

(i) As S → λ is a production, S λ. So λ is in L(G).

(ii) For all $n \geq 1$:

$$
\begin{aligned}
S &\underset{G}{\Rightarrow} 0S1 \\
&\underset{G}{\Rightarrow} 0^2S1^2 \text{ (S} \rightarrow \text{0S1 is applied)} \\
&\overset{*}{\underset{G}{\Rightarrow}} 0^nS1^n \text{ (S} \rightarrow \text{0S1 is applied for } (n-1) \text{ times)}
\end{aligned}
$$

$\underset{G}{\Rightarrow}$ 0^n1^n ($S \to \lambda$ is applied)

$\therefore$ $0^n1^n \in L(G)$, for $n \geq 0$.

Because $S \to 0S1$ is applied at every step except in the last step. In the last step we apply $S \to \lambda$. Hence $\{0^n1^n \mid n \geq 0\} \subseteq L(G)$

(iii) To show that $L(G) \subseteq \{0^n1^n \mid n \geq 0\}$, we start with w in L(G). The derivation of w starts with S.

(a) if $S \to \lambda$ is applied first, we get λ (i.e.) $w=\lambda$.

(b) if $S \to 0S1$ is applied first, then to derive a terminal string $S \to \lambda$ is applied at any stage.

(i.e.) $w = S \underset{G}{\Rightarrow} 0^n1^n$ for $n \geq 0$

(i.e.) $L(G) \subseteq \{0^n1^n \mid n \geq 0\}$

$\therefore$ $L(G) = \{0^n1^n \mid n \geq 0\}$

4.2 CHOMSKY CLASSIFICATION OF LANGUAGE

In the definition of a grammar (V_N, Σ, P, S), V_N and Σ are sets of symbols and $S \in V_N$. Therefore the chomsky classification of grammars is done on the basis of productions. They are:

(i) Type 0 – Unrestricted grammar

(or)

Phrase structure grammar

(ii) Type 1 – Context sensitive grammar

(or)

Context dependent grammar

(iii) Type 2 – Context free grammar

(iv) Type 3 – Regular grammar

The following table shows the general form of productions, type of automata used with examples.

Type	Grammar / Language	Form of production in grammar	Example	Automata (or) Accepting device
0	Unrestricted grammar (or) Phrase structure grammar	A grammar without any restrictions, with the following production form : $\phi A \psi \to \phi\alpha\psi$ A - variable ϕ - left context ψ - right context α - replacement string	(i) $ab\underline{A}bcd \to ab\underline{AB}bcd$ where *ab*-left context *bcd*-right context α= AB (ii) $A\underline{C} \to A$ where A - left context λ - right context $\alpha = \lambda$ (iii) $C \to \lambda$ λ - left and right context $\alpha = \lambda$	Turing Machine (TM)
1	Context sensitive grammar (or) Context dependent grammar	A production of the form $\phi A \psi \to \phi\alpha\psi$ is called type 1 production if $\alpha \neq \lambda$. In type 1 productions erasing of A is not permitted.	(i) $a\underline{A}bcD \to a\underline{bcD}bcD$ where *a* - left context *bcD* - right context A is replaced by $bcD \neq \lambda$ (ii) $A\underline{B} \to A\underline{bBC}$ where A - left context λ - right context $\alpha = bBC \neq \lambda$	Linear Bounded Automata (LBA)

			(iii) $A \to abA$ where λ - left & right context. $A = abA \neq \lambda$ (iv) $A \to \lambda$ is allowed but A does not appear on the right handside of any production of the grammar.	
2	Context free grammar	A production of the form $A \to \alpha$, where $A \in V_N$ and $\alpha \in (V_N \cup \Sigma)^*$ is called a type 2 grammar. In simple words left hand side has no left context or right context. $A =$ L.H.S. $\in V_N$ $\alpha =$ R.H.S $\in (V_N \cup \Sigma)^*$	(i) $S \to Aa$ (ii) $A \to a$ (iii) $B \to abc$ (iv) $A \to \lambda$ where L.H.S $\in V_N$ (Variables) and R.H.S. $\in (V_N \cup \Sigma)^*$ (Combination of variable & terminal)	Push Down Automata (PDA)
3	Regular grammar	A production of the form $A \to a$ or $A \to aB$ where $A, B \in V_N$ and $a \in \Sigma$ is called a type 3 production. A production $S \to \lambda$ is allowed in type 3, but in this case S does not appear on the right hand side of any production. $A =$ L.H.S. $\in V_N$ R.H.S $\in (V_N \cup a)$	(i) $S \to b \mid c$ (ii) $S \to bA$ (iii) $S \to a$ where L.H.S. $\in V_N$ (Variables) and R.H.S. belongs to either a single terminal or a terminal followed by one variable.	Finite Automata (FA)

Note :

(i) Let L_0 - type 0 language

L_{csl} - context sensitive language

L_{cfl} - context free language

L_{rl}-regular language

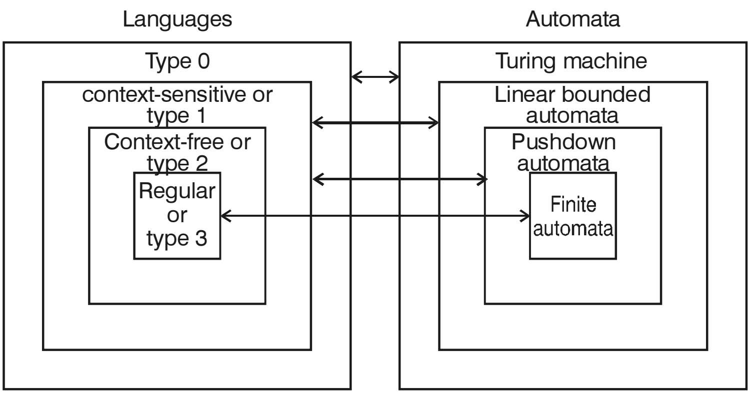
(ii) $L_{rl} \subseteq L_{cfl}$, $L_{cfl} \subseteq L_{csl}$, $L_{csl} \subseteq L_0$

(iii) From the above points :

$L_{rl} \subseteq L_{cfl} \subseteq L_{csl} \subseteq L_0$

4.3 LANGUAGES AND AUTOMATA

The following figure describes the relation between the four types of languages and automata : (discussed in the following chapters)



Languages and corresponding automata

4.4 SOLVED PROBLEMS

1. Find the highest type number which can be applied to the following grammar:

(a) $S \to Aa$, $A \to c \mid Ba$, $B \to abc$

(b) $S \to ASB \mid d$, $A \to aA$

(c) $S \to aS \mid ab$

Solution :

(a) $S \to Aa$, $A \to Ba$, $B \to abc$ - type 2.

(because productions are of the form $A \to \alpha$)

$A \to c$ - type 3 (because production of the form $A \to a$)

$\therefore$ the highest type number is 2.

(b) $S\rightarrow ASB$ - type 2 (i.e. $A\rightarrow\alpha$)

$S\rightarrow d$, $A\rightarrow aA$ - type 3 (i.e. $A\rightarrow a$, $A\rightarrow aB$)

$\therefore$ the highest type number is 2.

(c) $S\rightarrow aS$ - type 3 (i.e. $A\rightarrow aB$)

$S\rightarrow ab$ - type 2 (i.e. $A\rightarrow\alpha$)

$\therefore$ the highest type number is 2.

2. Identify the grammar type from the following productions:

(a) $S\rightarrow 0S1$

$S\rightarrow \lambda$

(b) $S\rightarrow aCa$

$C\rightarrow aCa \mid b$

(c) $S\rightarrow aS$

$S\rightarrow bS$

$S\rightarrow a \mid b$

(d) $S\rightarrow aSa \mid bSb$

$S\rightarrow a \mid b$

$S\rightarrow\lambda$

(e) $S\rightarrow A$

$A\rightarrow ab$

$A\rightarrow aAb$

$S\rightarrow Sc$

(f) $S\rightarrow 0SA_1 2$

$S\rightarrow 012$

$2A_1\rightarrow A_1 2$

$1A_1\rightarrow 11$

(g) $S\rightarrow aSBC \mid aBC$

$CB\rightarrow BC$

$aB\rightarrow ab$

$bB\rightarrow bb$

$bC \to bc$

$cC \to cc$

Solution :

(a) Context free grammar - type 2

(b) Context free grammar - type 2

(c) Regular grammar - type 3

(d) Context free grammar - type 2

(e) Context free grammar - type 2

(f) Context sensitive grammar - type 1

(g) Context sensitive grammar - type 1

Note for f & g :

Theorem : Every monotonic grammar G is equivalent to a type 1 grammar.

3. Test whether 001100, 001010, 01010 are in the language generated by the grammar $S \to 0S1|0A|0|1B|1$, $A \to 0A \mid 0$, $B \to 1B \mid 1$

(i.e.) $L(G)=\{0^m 1^n \mid m \neq n$ and atleast one of m and $n \geq 1\}$. Clearly $0^m \in L(G)$ and $1^n \in L(G)$ where $m, n \geq 1$.

Solution :

For the derivation of 001100, 001010 or 01010, the first production cannot be $S \to 0S1$. The other possible productions are $S \to 0A$ and $S \to 1B$. In these cases the resulting terminal strings are 0^n or 1^n. Therefore none of the given strings are in language generated by the given grammar.

4. State whether the following statement is true or false justify your answer with a proof or a counter example.

If G_1 and G_2 are equivalent, then they are of the same type.

Solution :

False. (Two Grammars of different types may generate the same language)

Example : $S \to aS|bS|a|b$ – L(G) – type 3

$S \to SS|aS|bS|a|b$–$L(G^1)$ - type 2

Then $L(G^1) = L(G)$ as the production $S \to aS|bS|a|b$ is in G also and $S \to SS$ does not generate any more string.

5. Construct (a) a context-sensitive but not context free grammar.

(b) a context-free but not regular grammar

(c) a regular grammar to generate $\{a^n \mid n \geq 1\}$

Solution :

(a) $S \to aS|B,\ aS \to aa,\ B \to a$

(b) $S \to AS|a,\ A \to a$

(c) $S \to aS|a$

6. Construct context free grammar for the given set.

$\{0^m 1^n | 1 \leq m \leq n\}$

Solution :

The required productions are:

$S \to 0S1$

$S \to 01$

$S \to 0A1$

$A \to 1A$

$A \to 1$

7. Construct regular grammars to generate the following:

(a) the set of all strings over $\{a,b\}$ ending in a.

(b) The set of all strings over $\{a,b\}$ beginning with a

(c) $\{a^{2n} \mid n \geq 1\}$

Solution:

(a) $S \to aS$

$S \to bS$

$S \to a$

(b) $S \to aS_1$

$S_1 \to aS_1$

$S_1 \to bS_1$

$S_1 \to a$

$S_1 \to b$

(c) $S \rightarrow aS_1$

$S_1 \rightarrow aS$

$S \rightarrow aS_2$

$S_2 \rightarrow a$

(i.e)

$S \Rightarrow aS_1$

$\Rightarrow aaS$ $(S_1 \rightarrow aS)$

$\Rightarrow aaaS_2$ $(S \rightarrow aS_2)$

$\Rightarrow aaaa$ $(S_2 \rightarrow a)$ (when n=2)

$S \Rightarrow aS_1$

$\Rightarrow aaS$ $(S_1 \rightarrow aS)$

$\Rightarrow aaaS_1$ $(S \rightarrow aS_1)$

$\Rightarrow aaa\underline{a}S$ $(S_1 \rightarrow aS)$

$\Rightarrow aaaaaS_2$ $(S \rightarrow aS_2)$

$\Rightarrow aaaaaa$ $(S_2 \rightarrow a)$ (when n=3)

8. Show that $G_1=(\{S\}, \{a,b\}, P_1, S)$, where $P_1=\{S \rightarrow aSb \mid ab\}$ is equivalent to $G_2=(\{S,A,B,C\}, \{a,b\}, P_2, S)$. Here P_2 consists of $S \rightarrow AC$, $C \rightarrow SB$, $S \rightarrow AB$, $A \rightarrow a$, $B \rightarrow b$.

Solution :

(i) From the production P_1, $L(G_1)$ is:

$S \Rightarrow aSb$

$\Rightarrow aaSbb$

$\Rightarrow a^n b^n$ for $n \geq 1$

$\therefore$ $L(G_1) = \{a^n b^n \mid n \geq 1\}$

(ii) From the production P_2, $L(G_2)$ is :

(a) $S \Rightarrow A\underline{C}$

$\Rightarrow A\underline{SB}$

$\Rightarrow A^{n-1} SB^{n-1}$

(b) $S \Rightarrow AB$

$\Rightarrow ab$

$\therefore$ $S \Rightarrow a^n b^n$ for $n \geq 1$

$\therefore$ $L(G_2) = \{a^n b^n | n \geq 1\}$

Hence for the given productions it has proved that $L(G_1) = \{a^n b^n | n \geq 1\} = L(G_2)$

9. For each production in a grammar G has some variable on its right-hand side, what can you say about L(G)

Solution : $L(G) = \phi$

Example : If G=({S}, {*a*}, {S–SS},S)

then $S \rightarrow SS$ in G has no terminal on the righthand side.

10. Construct a grammar to generate

$\{(ab)^n \mid n \geq 1\}$ $\{(ba)^n \mid n \geq 1\}$

Solution :

The required productions are:

$S \rightarrow S_1 \cup S_2$

$S \rightarrow S_1$

$S_1 \rightarrow abS_1$

$S_1 \rightarrow ab$

$S \rightarrow S_2$

$S_2 \rightarrow baS_2$

$S_2 \rightarrow ba$

11. Construct productions to generate the following:

(a) The set of all strings over $\{a,b\}$ ending in '*a*'

(b) The set of all strings over $\{a,b\}$ beginning with '*a*'.

Solution :

(a) As per the given condition, strings has to end with '*a*' but it may start with either '*a*' (or) '*b*'.

Productions :

$S \rightarrow aS$

$S \rightarrow bS$

$S \rightarrow a$

(i.e.) $S \rightarrow aS \mid bS \mid a$

(b) As per the given constraint, all strings has to begin with '*a*' but it may end with either *a* (or) *b*.

Productions :

$S \rightarrow aS_1$

$S_1 \rightarrow aS_1$

$S_1 \rightarrow bS_1$

$S_1 \rightarrow a$

$S_1 \rightarrow b$

(i.e.) $S \rightarrow aS_1$

$S_1 \rightarrow aS_1 \mid bS_1 \mid a \mid b$

12. Construct context free grammars to accept the following languages.

a. {w | w starts and ends with the same symbol}

$S \rightarrow 0A0 \mid 1A1$

$A \rightarrow 0A \mid 1A \mid e$

b. {w | |w| is odd}

$S \rightarrow 0A \mid 1$

$A \rightarrow 0S \mid 1S \mid e$

c. {w | |w| is odd and its middle symbol is 0}

$S \rightarrow 0 \mid 0S0 \mid 0S1 \mid 1S0 \mid 1S1$

d. $\{0^n1^n \mid n>0\}$ U $\{0^n1^{2n} \mid n>0\}$

$S \rightarrow 0A1 \mid 0B11$

$A \rightarrow 0A1 \mid e$

$B \rightarrow 0B11 \mid e$

e. $\{0^i1^j2^k \mid i!=j \text{ or } j!=k\}$

$S \rightarrow AC \mid BC \mid DE \mid DF$

$A \rightarrow 0 \mid 0A \mid 0A1$

$B \rightarrow 1 \mid B1 \mid 0B1$

$C \rightarrow 2 \mid 2C$

$D \rightarrow 0 \mid 0D$

$E \rightarrow 1 \mid 1E \mid 1E2$

$F \rightarrow 2 \mid F2 \mid 1F2$

f. Binary strings with twice as many 1s as 0s.

$S \rightarrow e \mid 0S1S1S \mid 1S0S1S \mid 1S1S0S$

13. Explain why the grammar below is ambiguous.

$S \rightarrow 0A \mid 1B$

$A \rightarrow 0AA \mid 1S \mid 1$

$B \rightarrow 1BB \mid 0S \mid 0$

The grammar is ambiguous because we can find strings which have multiple derivations:

Derivation 1:

$$\begin{aligned} S &\Rightarrow 0A \\ &\Rightarrow 0\ 0AA \\ &\Rightarrow 00\ 1S\ 1 \\ &\Rightarrow 001\ 1B\ 1 \\ &\Rightarrow 0011\ 0\ 1 \end{aligned}$$

Derivation 2:

$$\begin{aligned} S &\Rightarrow 0A \\ &\Rightarrow 0\ 0AA \\ &\Rightarrow 00\ 1\ 1S \\ &\Rightarrow 0011\ 0A \\ &\Rightarrow 00110\ 1 \end{aligned}$$

14. Construct a context-free grammar for generating the language $L = \{a^n b^n \mid n \geq 1\}$. **(Nov./Dec 2004)**

Solution :

Grammar $G=(\{S\}, \{a, b\}, P, S)$, where

$P = \{S \rightarrow aSb \mid ab\}$

$$\begin{aligned} S &\Rightarrow aSb \\ &\Rightarrow aaSbb \ (S \rightarrow aSb) \\ &\Rightarrow aaabbb \ (S \rightarrow ab) \\ &\Rightarrow a^n b^n \text{ for } n \geq 1 \end{aligned}$$

15. Find the language generated by the grammar ($S \rightarrow aSb$, $S \rightarrow ab$). **(Apr./May 2005)**

Solution:

$L = \{a^n b^n \mid n \geq 1\}$

$$
\begin{aligned}
S &\Rightarrow aSa \\
&\Rightarrow aabb \ (S \rightarrow ab) \\
&\Rightarrow a^n b^n
\end{aligned}
$$