Content free grammar:-

- Grammers are used to generate the words of a language and to determine whether a word is in a language.

Formal Def^n of CFG:-

- A content free grammar (CFG) is defined by 4-tuples (V,T,P,S) where,
  - V = set of variables / set of non terminals
  - T = set of terminal symbols
  - P = set of productions (rules)
  - S = start symbol, $S \in V$.

Four components in CFG:-

1. There is a finite set of symbols that form the strings of language being defined. We call this alphabet the terminal or terminal symbol.
2. There is a finite set of variables, also called non terminal symbols. Each variable represent a language i.e. set of strings.
3. One of the variable represents the language being defined. It is called the start symbol.
4. There is a finite set of productions or rules that represent the recursive definition of the language.

Date: Page:

Each production consists of:

a. A variable that is being defined by the production. This is called head of production.

b. The production symbol $\longrightarrow$

c. A string of zero or more terminals and variables. This string is called the body of the production, represents one way to form the string in the language of the head.

Note:-

The variable symbols are represented by capital letters.
The terminal symbols are represented by lowercase letter.
One of the variable is desigred as a start variable,
It usually occurs on the left hand side of the production rule.

eg:-

$S \longrightarrow \epsilon$

$S \longrightarrow 0S1$

This is a CFG defining the grammar of all the strings with equal no. of 0's followed by equal no. of 1's.

Here;

Two rules define the production P,
$\epsilon, 0, 1$ are the terminal defining T,
S is a variable symbol defining V.
& S is a start symbol from where production starts.

Scanned with CamScanner

Compact notation of the productions:-

- It is convenient to think of a production as "belonging" to the variable of head. We may use the terms like A-production or production of A to the production whose head is A. We write the production ~~of~~ for a grammar by listing each variable once & listing all the variables, separated by |. Which is called compact notation.

eg. CFG representing the language over $\Sigma = \{a, b\}$ which is palindrome language.

$S \rightarrow \epsilon | a | b$; compact notation.

$S \rightarrow aSa$

$S \rightarrow bSb$

Here, the first production is written in Compact notation form. The detail of the production looks like:

$S \rightarrow \epsilon$

$S \rightarrow a$

$S \rightarrow b$

Date:________ Page:________

Derivation:-

CFG generates string according to the following process, called a derivation.

a) We start by writing down the start symbol of the CFG, S.

b) At each step of the derivation, we may replace any non-terminal symbol of the string generated so far by the right hand side of any production that has the symbol on the left.

c) The process ends when it is impossible to apply a step of the type described in (b).

⇒ If the string at the end of this process consists entirely of terminals, it is a string in the language generated by the grammar called yield.

There are two approaches of derivation

1) Body to head (Bottom Up) approach
2) Head to body (Top down) "

Body to head:-

⇒ Here, we take strings known to be in the language of each of the variables of the body, concatenate them, in the proper order, with any terminals appearing in the body, the resulting string is the language of the variable in the head.

Consider grammar,

$S \rightarrow S+S$

$S \rightarrow S/S$

$S \rightarrow (S)$

$S \rightarrow S-S$

$S \rightarrow S*S$

$S \rightarrow a$

here; given $a+(a*a)/a-a$

now, applying body to head approach;

S.N	String inferred	variable	Production	String used
1.	a	S	$S \rightarrow a$	—
2.	$a*a$	S	$S \rightarrow S*S$	string 1
3.	$(a*a)$	S	$S \rightarrow (S)$	string 2
4.	$(a*a)/a$	S	$S \rightarrow S/S$	string 3 & 1
5.	$(a*a)/a-a$	S	$S \rightarrow S-S$	string 4 & 1
6.	$a+(a*a)/a-a$	S	$S \rightarrow S+S$	string 5 & 1

Thus, in this approach we start with any terminal appearing in the body & use the available rules from body to head.

Date: Page:

Head to body:-

Here, we use production from head to body. We expand the start symbol using a production, whose head is the start symbol. Here we expand the resulting string until all strings of terminal are obtained. Here we have two approaches.

⇒ Leftmost Derivation:-

Here, leftmost symbol (variable) is replaced first.

Rightmost Derivation:-

Here, rightmost symbol is replaced first.

eg:- Consider the previous example of deriving string, $a+(a*a)/a-a$ with the above grammar

LMD:-

	rules
$S \rightarrow \underline{S}+S$	$S \rightarrow S+S$
$S \rightarrow a+S$	$S \rightarrow a$
$S \rightarrow a+S-S$	$S \rightarrow S-S$
$S \rightarrow a+S/S-S$	$S \rightarrow S/S$
$S \rightarrow a+(S)/S-S$	$S \rightarrow (S)$
$S \rightarrow a+(S*S)/S-S$	$S \rightarrow S*S$
$S \rightarrow a+(a*S)/S-S$	$S \rightarrow a$
$S \rightarrow a+(a*a)/S-S$	$S \rightarrow a$
$S \rightarrow a+(a*a)/a-S$	$S \rightarrow a$
$S \rightarrow a+(a*a)/a-a$	$S \rightarrow a$

RMD:-

	Rules
$S \to S-S$	$S \to S-S$
$S \to S-a$	$S \to a$
$S \to S+S-a$	$S \to S+S$
$S \to S+S/S-a$	$S \to S/S$
$S \to S+S/a-a$	$S \to a$
$S \to S+(S)/a-a$	$S \to (S)$
$S \to S+(S*S)/a-a$	$S \to S*S$
$S \to S+(S*a)/a-a$	$S \to a$
$S \to S+(a*a)/a-a$	$S \to a$
$S \to a+(a*a)/a-a$	$S \to a$

Direct derivation:-

$\alpha_1 \to \alpha_2$;

- If α_2 can be derived directly from α_1, then it is called direct derivation.

$\alpha_1 \xrightarrow{*} \alpha_2$;

- If α_2 can be derived from α_1 with zero or more steps of the derivation, then it is just derivation.
- also called sentential form for each results derived.

$S \overset{*}{\Rightarrow} \alpha$

eg

$S \to asa/ab/a/b/\epsilon$

$S \to ab$; direct derivation.

Sentential form

$$\left.\begin{array}{l} S \to asa \\ S \to aasaa \\ S \to aabaa \end{array}\right\} \text{just derivation}$$

Derivation Tree / Parse tree :-

- Parse tree is a tree representation of strings of terminals using the productions defined by the grammar.
- A parse tree pictorially shows how the start symbol of a grammar derives a string in the language.
- Formally, given CFG $G = (V, T, P, S)$, a parse tree is a tree having following properties.
  - The root is labelled by the start symbol.
  - Each interior node of parse tree are variables.
  - Each leaf node of parse tree is labelled by a terminal symbol or ϵ.

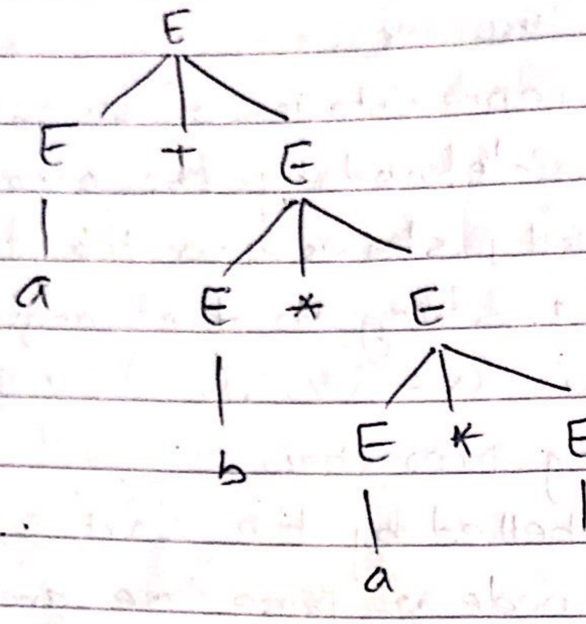
eg Let $G = (V, T, P, S)$ be a grammar where

$V = \{E\}$
$T = \{+, *, a, b\}$
$S = E$
P consists of
$E \rightarrow E+E \mid E*E \mid a \mid b$

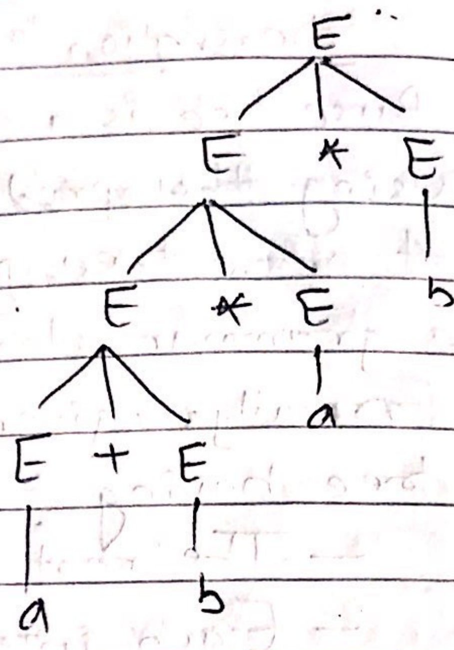
Derive $w = a+b*a*b$ using LMD & RMD. Also construct respective derivation tree.

Soln:-

LMD	RMD
$S \rightarrow E$	$S \rightarrow E$
$\rightarrow E+E$	$\rightarrow E*E$
$\rightarrow a+E$	$\rightarrow E*b$
$\rightarrow a+E*E$	$\rightarrow E*E*b$
$\rightarrow a+b*E$	$\rightarrow E*a*b$
$\rightarrow a+b*E*E$	$\rightarrow E+E*a*b$
$\rightarrow a+b*a*E$	$\rightarrow E+b*a*b$
$\rightarrow a+b*a*b$ //	$\rightarrow a+b*a*b$ //



LMD parse tree



RMD parse tree

Exercise:-

a) Consider the grammar G.

$S \longrightarrow A1B$

$A \longrightarrow 0A \mid \epsilon$

$B \longrightarrow 0B \mid 1B \mid \epsilon$

1) Construct the parse tree for 00101

LMD

$S \longrightarrow A1B$
$\rightarrow 0A1B$
$\rightarrow 00A1B$
$\rightarrow 00\epsilon 1B$
$\rightarrow 0010B$
$\rightarrow 00101B$
$\rightarrow 00101\epsilon$
$\rightarrow 00101$

RMD

$S \rightarrow A1B$
$\rightarrow A10B$
$\rightarrow A101B$
$\rightarrow A101\epsilon$
$\rightarrow 0A101$
$\rightarrow 00A101$
$\rightarrow 00\epsilon 101$
$\rightarrow 00101$

2) Construct a parse tree for 1001

3) Construct a parse tree for 00011

Date: Page:

Q: The production of CFG is

$S \rightarrow 0B \mid 1A$

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

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

for the string 00110101, find LMD and RMD. Also draw respective parse tree

Ambiguity in Grammar:-

A grammar $G = (V, T, P, S)$ is said to be ambiguous if there is a string $w \in L(G)$ for which we can derive two or more distinct derivation tree rooted at S and yielding w. In other words, a grammar is ambiguous if there exists more than one leftmost derivation or more than one rightmost derivation for the same string.

example:-

$S \rightarrow AB / aaB$

$A \rightarrow a / Aa$

$B \rightarrow b$

for any string aab;
We have two leftmost derivation as,

$S \rightarrow AB$
$\rightarrow AaB$
$\rightarrow aaB$
$\rightarrow aab$

Also, $S \rightarrow aaB$
$\rightarrow aab$

Simplification of CFG :-

⇒ There are mainly 3 simplifications of CFGs.
1) Elimination of epsilon production
2) ,, ,, Unit ,,
3) ,, ,, Useless symbols.

Removal of Epsilon (ϵ) - production | null production:

⇒ A production of the form
non-terminal $\longrightarrow$ empty string is called null production.

eg:- $S \longrightarrow \epsilon$, $A \longrightarrow \epsilon$ etc.

⇒ To eliminate ϵ-production(s) from a CFG, we first find out nullable non-terminal(s) of CFG. A non terminal is nullable if it derives an epsilon in zero or more steps.

i.e. $A \xrightarrow{*} \epsilon$

then A is nullable.

⇒ Then we search for occurance of the non-terminals that generates empty string. i.e. in all other production rules.

⇒ If such occurance of the non-terminal exists, then we replace that non terminal with the empty string and add the resulting production to the grammar.

⇒ After adding the new production to the grammar, we discard the null production from the grammar. The process is continued until all the null productions are eliminated.

eg:- Consider the grammar

$S \rightarrow aA$

$A \rightarrow b \mid \epsilon$

Remove the null production.

Soln:- Here, the null production is

$A \rightarrow \epsilon$

Since, the occurance of A exist in the production

$S \rightarrow aA$.

Therefore, we replace A by ϵ and add the resulting production rule to the grammar.

$S \rightarrow aA$

$S \rightarrow a\epsilon$

$S \rightarrow a$

- After adding $S \rightarrow a$ to the grammar, the grammar becomes,

$S \rightarrow aA \mid a$

$A \rightarrow b$

Which is desired simplified grammar.

eg Consider the following grammar G.

classnote

$$S \rightarrow ABAC$$
$$A \rightarrow aA \mid \epsilon$$
$$B \rightarrow bB \mid \epsilon$$
$$C \rightarrow c$$

Q) Remove the ϵ-production from above grammar.

Soln:- Here, null production are:

$A \longrightarrow \epsilon$ and $B \longrightarrow \epsilon$

for $A \longrightarrow \epsilon$

∵ the occurance of A exist, in the production

~~ABAC~~ $S \longrightarrow ABAC$

∴ We replace 'A' by 'ϵ' in all ways

$S \rightarrow ABAC$
$\rightarrow \epsilon BAC$
$\rightarrow BAC$

$S \rightarrow ABAC$
$\rightarrow AB\epsilon C$
$\rightarrow ABC$

$S \rightarrow ABAC$
$\rightarrow \epsilon B\epsilon C$
$\rightarrow BC$

and

$A \rightarrow aA$
$\rightarrow a\epsilon$
$\rightarrow a$

Adding new productions to grammar;

$S \rightarrow ABAC \mid ABC \mid BAC \mid BC$
$A \rightarrow \cancel{BBAC}\ aA \mid a$
$B \rightarrow bB \mid \epsilon$
$C \rightarrow c$

Contd - - -

Contd..

we apply rule in the new production we achieved just

for $B \to \epsilon$

$S \to ABAC$
$\to A\epsilon AC$
$\to AAC$

$S \to ABC$
$\to A\epsilon C$
$\to AC$

$S \to BAC$
$\to \epsilon AC$
$\to AC$

$S \to BC$
$\to \epsilon C$
$\to C$

and $B \to bB$
$\to b\epsilon$
$\to b$

∴ final grammar is

$S \to ABAC \mid ABC \mid BAC \mid BC \mid AAC \mid AC \mid C$
$A \to aA \mid a$
$B \to bB \mid b$
$C \to c$

eg

Consider the grammar: * delete

$S \to ABC$
$A \to BB \mid \epsilon$
$B \to CC \mid a$
$C \to AA \mid b$

Soln: here,

$A \to \epsilon$, A is nullable
$C \to AA \xrightarrow{*} \epsilon$, C is nullable
$B \to CC \xrightarrow{*} \epsilon$, B is nullable

$S \longrightarrow ABC \longrightarrow^{*} \epsilon$, S is nullable

for $A \rightarrow \epsilon$,

$S \rightarrow \underline{A}BC$

$\rightarrow BC$

now Adding these to grammar;

$S \longrightarrow ABC \mid BC$

$A \longrightarrow BB$

$B \rightarrow CC \mid a$

$C \rightarrow AA \mid b$

for $C \rightarrow AA$,

$C \longrightarrow AA$	$C \rightarrow AA$	$C \rightarrow AA$
$\longrightarrow \epsilon A$	$\rightarrow A\epsilon$	$\rightarrow \epsilon\epsilon$
$\longrightarrow A$	$\longrightarrow A$	$\rightarrow \epsilon$

adding these to grammar.

$S \longrightarrow ABC \mid BC$

$A \longrightarrow BB$

$B \longrightarrow CC \mid a$

$C \rightarrow AA \mid A \mid \epsilon \mid b$

$\because C \longrightarrow \epsilon$,

$S \longrightarrow ABC$	$S \longrightarrow BC$	$B \longrightarrow CC$	$B \rightarrow CC$
$\rightarrow AB\epsilon$	$\rightarrow B\epsilon$	$\rightarrow C\epsilon$	$\rightarrow \epsilon\epsilon$
$\rightarrow AB$	$\longrightarrow B$	$\rightarrow C$	$\rightarrow \epsilon$

now, adding these to grammar.

$S \to ABC \mid BC \mid AB \mid B$

$A \to BB$

$B \to CC \mid C \mid \epsilon \mid a$

$C \to AA \mid A \mid b$

again,
for, $B \to \epsilon$,

$S \to ABC \to A\epsilon C \to AC$ $\quad$ $S \to BC \to \epsilon C \to C$ $\quad$ $S \to AB \to A\epsilon \to A$ $\quad$ $S \to B$, $S \to \epsilon$

$A \to BB \to B\epsilon \to B$ $\quad$ $A \to BB \to \epsilon B \to B$ $\quad$ $A \to BB \to \epsilon\epsilon \to \epsilon$

now, the grammar becomes.

$S \to ABC \mid BC \mid AB \mid B \mid AC \mid C \mid A \mid \epsilon$

$A \to BB \mid B \mid \epsilon$ already eliminated. ∴ eliminate this

$B \to CC \mid C \mid a$

$C \to AA \mid A \mid b$

for $S \to \epsilon$, no occurance of non terminal that generate empty string. i.e. other production rules.

∴ final CFG:- $S \to ABC \mid BC \mid AB \mid B \mid AC \mid C \mid A$

$A \to BB \mid B$

$B \to CC \mid C \mid a$

$C \to AA \mid A \mid b$

eg Remove ε-productions for each of grammar :

$$S \longrightarrow AB$$
$$A \longrightarrow aAA \mid \epsilon$$
$$B \longrightarrow bBB \mid \epsilon$$

Eliminating unit production:-

- A production of the form
  non-terminal $\longrightarrow$ non-terminal is called unit production.

  eg:- $S \longrightarrow A$, $A \longrightarrow B$ etc.

- To eliminate such production rules from the given CFG, we first find all the unit production of the form $\alpha \longrightarrow \beta$, where $\alpha, \beta \in V$. Then we search for a production of the form $\beta \longrightarrow a$, where $\beta \in V$ and $a \in T$.
- If such production rule exists then we replace the right hand side non-terminal of the unit production with respective terminal.
  ie. we add the production $\alpha \longrightarrow a$, where $\alpha \in V$ and $a \in T$ to the grammar and discard $\alpha \longrightarrow \beta$ from the grammar.
- We continue the process until all the unit productions are eliminated from the CFG.

eg Consider the CFG as:

$S \longrightarrow AB$
$A \longrightarrow a$
$B \longrightarrow C|b$
$C \longrightarrow D$
..

Date: Page:

$D \rightarrow E$

$E \rightarrow a$

Remove the unit production.

Soln:- Here, the unit production of the given grammar are:-

$B \rightarrow C$

$C \rightarrow D$

$D \rightarrow E$

- Now, we have to search for a production ~~gote~~ rule suchthat either of C, D or E gives a terminal symbol. Among these unit productions, the right hand side non-terminal of unit production $D \rightarrow E$ gives a terminal symbol 'a'.
  i.e. $E \rightarrow a$ exists in the given grammar.
- Now we replace E by a in the unit production $D \rightarrow E$ such that a new production $D \rightarrow a$ is added to the grammar and $D \rightarrow E$ is discarded from the given grammar.

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow C \mid b$

$C \rightarrow D$

$D \rightarrow a$

$E \rightarrow a$

as $D \rightarrow a$

also, $C \rightarrow D$ $\therefore C \rightarrow a$

now $S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow C \mid b$

$C \rightarrow a$

$D \rightarrow a$

$E \rightarrow a$

Scanned with CamScanner

Date: Page:

$\therefore C \to a$

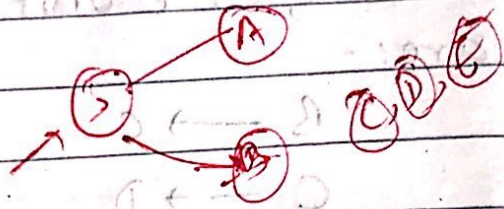
also, $B \to C \quad \therefore B \to a$

– final grammar after removing unit production is:

$S \to AB$

$A \to a$

$B \to a \mid b$



$\cancel{C \to a}$

$\cancel{D \to a}$

$\cancel{E \to a}$

$\therefore$ C, D, E are unreachable ie. there is no way we can reach symbols C, D, E from the start symbol.

Eg: Consider the CFG as:

$S \to AA$

$A \to B \mid BB$

$B \to abB \mid b \mid bb$

Remove the unit production.

Soln:- The unit production of above grammar is $A \to B$

– Now, we have to search for production rule such that B gives a terminal symbol.

Here, the production rule $A \to B$, gives a terminal symbol

ie $B \to b$

Also, $B \to bb$

now,

$$A \rightarrow b \mid bb$$

∴ The simplified grammar is

$$S \rightarrow AA$$

$$A \rightarrow b \mid bb \mid BB \mid abB$$

also

$$B \rightarrow abB \mid b \mid bb$$

eg Remove Unit productions from the grammar whose production rule is given by

P: $S \rightarrow XY$

$X \rightarrow a$

$Y \rightarrow Z \mid b$

$Z \rightarrow M$

$M \rightarrow N$

$N \rightarrow a$

Q:- Consider the CFG G given below. Find a CFG G' with no ϵ-production and no unit productions.

⇒

$S \to ABA$

$A \to aA | \epsilon$

$B \to bB | \epsilon$

Soln:- Here,

$A \to \epsilon$

$B \to \epsilon$

are null productions

⇒ for $A \to \epsilon$

The occurance of A exists in the production as

$S \to \underline{A}BA$ $\quad$ $S \to AB\underline{A}$ $\quad$ $S \to \underline{A}B\underline{A}$

$\to BA$ $\quad$ $\to AB$ $\quad$ $\to B$

also,

$A \to a\underline{A}$

$\to a$

∴ After removing $A \to \epsilon$ production, we add these new productions to grammar.

$S \to ABA | BA | AB | B$

$A \to aA | a$

$B \to bB | \epsilon$

$\Rightarrow$ for $B \rightarrow \epsilon$

$S \rightarrow ABA$ $\quad$ $S \rightarrow BA$ $\quad$ $S \rightarrow AB$ $\quad$ $S \rightarrow B$

$\rightarrow A\epsilon A$ $\quad$ $\rightarrow \epsilon A$ $\quad$ $\rightarrow A\epsilon$ $\quad$ $\rightarrow \epsilon$

$\rightarrow AA$ $\quad$ $\rightarrow A$ $\quad$ $\rightarrow A$

also

$B \rightarrow bB$

$\rightarrow b\epsilon$

$\rightarrow b$

$\therefore$ After removing $B \rightarrow \epsilon$, we add these new productions to grammar.

$S \rightarrow ABA \mid BA \mid AB \mid B \mid AA \mid A \mid \epsilon$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

for $S \rightarrow \epsilon$, no occurence of S, so simply remove $S \rightarrow \epsilon$

$\therefore$ final production after removing null productions are :-

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid A \mid B$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

again, in the above CFG, unit productions are:-

$S \rightarrow A$

$S \rightarrow B$

⇒ Now we have to search a production rule such that either of ~~A, B~~ A,B gives a terminal symbol.

i.e. $B \rightarrow b$

$A \rightarrow a$

exists in the given grammar.

⇒ Now we replace 'B' by 'b' and 'A' by 'a' in the unit production

$S \rightarrow A$ & $S \rightarrow B$

Such that $S \rightarrow a$ & $S \rightarrow b$ is added to the grammar and $S \rightarrow B$ and $S \rightarrow A$ is discarded from the grammar.

⇒ ∴ The final CFG after removal of unit production is:

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid a \mid b$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

eg Consider the CFG G given below. Find a CFG G' with no ε-productions & no unit productions:

$S \rightarrow aSa \mid bSb \mid \epsilon$

$A \rightarrow aBb \mid bBa$

$B \rightarrow aB \mid bB \mid \epsilon$

Eliminating useless symbols :-

⇒ A symbol X is said to be useful if

i) X is generating, If X is of the form

$$X \xrightarrow{*} w, \quad w \in T$$

i.e. If X is generating, It gives a terminal symbol or symbols after using series of production rule.

ii) X is reachable from start symbol, If X is of the form

$$S \xrightarrow{*} \alpha X \beta$$

⇒ A symbol that doesnot satisfy these two conditions is called useless symbol.

⇒ To eliminate useless symbols from the grammar, we first find all the generating symbols in the given grammar. i.e. all the symbols that generates a terminal symbol or terminal strings. If any symbol doesnot terminate then discard this symbol from the grammar.

Date: Page:

⇒ There may be some symbols that generates terminal symbol or string but they are not reachable from the start symbol. Such symbols have no roles in string generating. Therefore, we have to discard this as well.

⇒ To eliminate the non-reachable symbol, we draw a dependency graph using these symbols.

- Any symbol that remains unconnected in a graph is non-reachable symbol and we discard it from the grammar.

⇒ Thus after eliminating the non-generating symbols and unreachable symbols, we shall have only the useful symbol left.

Remember that:-

- Whenever we have to simplify any CFG, we first, make sure the CFG is:
  - ϵ-production free
  - unit-production free
- Then only we apply elimination of useless symbol.
- We must follow this order of processing for simplification of CFG.

eg: Consider the grammar G;

$S \rightarrow aB \mid bX$

$A \rightarrow Bad \mid bSX \mid a$

$B \rightarrow aSB \mid bBX$

$X \rightarrow SBd \mid aBX \mid ad$

Eliminate useless symbols :-

Soln:- Here,

S, A, X are only generating symbols. Therefore remove B from the grammar. Since B is non generating.

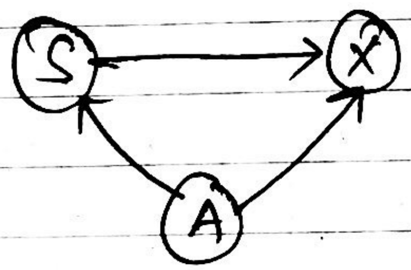
⇒ The new grammar after removing non-generating symbol is

$S \rightarrow bX$

$A \rightarrow bSX \mid a$

$X \rightarrow ad$

⇒ now, to Remove the non-reachable symbol,



⇒ It is seen clearly that from start symbol 'S' we cannot reach A. So A is unreachable

∴ the simplified grammar is;

$S \to bX$

$X \to ad$

Q. Find a CFG equivalent to

$S \to AB \mid CA$

$A \to a$

$B \to BC \mid AB$

$C \to aB \mid b$

with no useless symbols.

Soln:- Here, A and C are generating symbols, since they can directly generate terminal symbol.

Also, S can produce terminals. Since $S \to CA$, where both C & A can produce terminals.

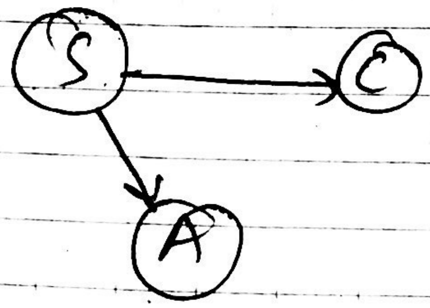
- But B does not produce any terminal symbol. Therefore B is non-generating. Thus we eliminate B from the CFG. The remaining CFG becomes,

$S \to CA$

$A \to a$

$C \to b$

- Now for removing un-reachable symbol, we draw the dependency graph as,



In the above CFG, there is no any symbol which is unreachable.

$\therefore$ final CFG is

$$S \rightarrow eA$$
$$A \rightarrow a$$
$$C \rightarrow b$$

Q: Simplify the following grammar:

$$S \rightarrow a \mid aA \mid B \mid C$$
$$A \rightarrow aB \mid \epsilon$$
$$B \rightarrow aA$$
$$C \rightarrow aCD$$
$$D \rightarrow ddd$$

Soln:

For removing ϵ-production,

for, $A \rightarrow \epsilon$

$$S \rightarrow aA$$
$$\rightarrow a$$

and

$$B \rightarrow aA$$
$$\rightarrow a$$

$\therefore$ After eliminating null production $A \rightarrow \epsilon$, grammar will be:

$$S \rightarrow a \mid aA \mid B \mid C$$
$$A \rightarrow aB \mid a$$
$$B \rightarrow aA \mid a$$

Scanned with CamScanner

$C \rightarrow aCD$

$D \rightarrow ddd$

For removing Unit production;

$S \rightarrow B$

$S \rightarrow C$

~~Basic unit pair~~

~~$(S,S), (B,B), (C,C), (A,A), (D,D)$~~

now,

for $S \rightarrow B$ ∴ ~~(S,B) is a unit pair~~.

~~$S \rightarrow B \rightarrow aA$~~

~~$S \rightarrow B \rightarrow a$~~

∴ $S \rightarrow a \mid aA \mid C$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

$C \rightarrow aCD$ // Here no terminal found for C

$D \rightarrow ddd$

for $S \rightarrow C$ ∵ ~~(S,C) is a unit pair~~

~~$S \rightarrow C \rightarrow aCD$~~

i.e. such case if no terminal found then to remove unit production keep it at C

∴ $S \rightarrow a \mid aA \mid \cancel{aCD}\ aCD$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

$C \rightarrow aCD$

$D \rightarrow ddd$

We can eleminate $C \rightarrow aCD$ ∵ $S \rightarrow C$ (unit production) and $C \rightarrow aCD$ which is non terminal generating for removing useless symbols:

Here, S, B, D can directly generate terminal symbols or strings. So S, B, D are generating symbols.

A is also generating because in production,

$A \rightarrow aB$, B is generating.

But,

C is non generating. Hence remove all the production of C.

∴

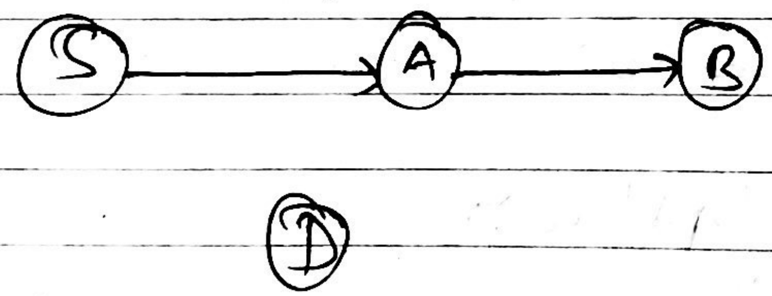
$S \rightarrow a \mid aA$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

$D \rightarrow ddd$

for non-reachable,



Here, clearly D is non-reachable. Hence final CFG is;

$S \rightarrow a \mid aA$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

Date:________ Page:______

Q. Convert the grammar To GNF

$S \rightarrow abaSa \mid aba$

⇒ If we introduce new variables A & B & productions as $A \rightarrow a$, $B \rightarrow b$ & substitute into grammar as.

$S \rightarrow aBASA \mid aBA$

$A \rightarrow a$, $B \rightarrow b$ which is in GNF

Q.

$S \rightarrow AB$

$A \rightarrow aA / bB \mid b$

$B \rightarrow b$.

⇒ We can replace A in S by aA or bB or b

∴ $S \rightarrow aAB \mid bBB \mid bB$

$A \rightarrow aA \mid bB \mid b$

$B \rightarrow b$

Which is in GNF

Normal forms of CFG:-

① Chomsky Normal form (CNF):-
② Greibach Normal form (GNF)

Chomsky Normal form (CNF):-

- A CFG $G = (V, T, P, S)$ is said to be in chomsky normal form (CNF) if every production in G are in one of the two forms.

$A \longrightarrow BC$ and
$A \longrightarrow a$

Where $A, B, C \in V$ and $a \in T$.

- Thus a grammar in CNF is one which should not have;

  ϵ-production
  Unit ,,
  Useless ,,

$S \rightarrow \epsilon$
$S \rightarrow aS$ ✗
only (LHS)

Used for Normalizing to be used in algorithm like CYK. (for parsing)

Eg: $A \longrightarrow a$
$S \longrightarrow AB$ etc are in CNF.

not now
i/o Later

Conversion to CNF:-

eg: Convert the following CFG into CNF.

$S \longrightarrow aAB$
$A \longrightarrow aA$
$B \longrightarrow bBC$
$C \longrightarrow bc$

Soln:- Here, the above CFG is already simplified.

Page:

Cycle 1 :-

i) Consider the production, $S \longrightarrow aAB$. we can rewrite this production as $S \longrightarrow V_1 AB$, where $V_1 \longrightarrow a$ and V_1 is a new non-terminal.

ii) Consider the production, $A \longrightarrow aA$, we can rewrite this production as $A \longrightarrow V_1 A$ ($\because V_1 \longrightarrow a$ from (i))

iii) Consider the production $B \longrightarrow bBC$, we can re-write the production as $B \longrightarrow V_2 BC$, where $V_2 \longrightarrow b$ and V_2 is a new non terminal.

iv) Consider the production $C \longrightarrow bc$, we can re-write this production as.

$C \longrightarrow V_2 V_3$, where $V_2 \longrightarrow c$ & $V_3 \longrightarrow c$

where V_3 is a new non-terminal.

from (i), (ii), (iii), (iv) we get

$S \longrightarrow V_1 AB$

$V_1 \longrightarrow a$

$A \longrightarrow V_1 A$

$B \longrightarrow V_2 BC$

$V_2 \longrightarrow b$

$C \longrightarrow V_2 V_3$

$V_3 \longrightarrow c$

Cycle 2 :-

v) Productions $V_1 \longrightarrow a$, $A \longrightarrow V_1 A$, $V_2 \longrightarrow b$, $C \longrightarrow V_2 V_3$ are already in CNF.

vi) Consider the production ~~$S \rightarrow V_1 AB$~~ $S \rightarrow V_1 AB$
we can rewrite this production as $S \rightarrow V_1 V_4$
where $V_4 \rightarrow AB$ and V_4 is a new non-terminal.

vii) Consider the production $B \rightarrow V_2 BC$, we can re-write this production as $B \rightarrow V_2 V_5$
where $V_5 \rightarrow BC$ and V_5 is a new non-terminal.

∴ the final CNF is as,

$$S \longrightarrow V_1 V_4$$
$$V_4 \longrightarrow AB$$
$$V_1 \longrightarrow a$$
$$A \longrightarrow V_1 A$$
$$B \longrightarrow V_2 V_5$$
$$V_2 \longrightarrow b$$
$$V_5 \longrightarrow BC$$
$$C \longrightarrow V_2 V_3$$
$$V_3 \longrightarrow c$$

Q Convert the following CFG into CNF

$$S \longrightarrow aAbB$$
$$A \longrightarrow aA \mid a$$
$$B \longrightarrow bB \mid b$$

Soln:- Given CFG is already simplified

Cycle I :

i) Production $A \rightarrow a$ and $B \rightarrow b$ are already in CNF.

ii) Consider the production $S \longrightarrow aAbB$, we can rewrite as, $S \longrightarrow V_1 A V_2 B$ where $V_1 \rightarrow a$, $V_2 \rightarrow b$

Date: Page:

and V_1 & V_2 are new non-terminals.

iii) Consider the production $A \longrightarrow aA$, we can re-write as,

$A \longrightarrow V_1 A$

iv) Consider the production $B \longrightarrow bB$, we can re-write as,

$B \longrightarrow V_2 B$

Thus, from i), ii), iii) & iv)

$S \longrightarrow V_1 A V_2 B$	$V_1 \longrightarrow a$
$A \longrightarrow V_1 A \mid a$	$V_2 \longrightarrow b$
$B \longrightarrow V_2 B \mid b$	

cycle 2 :-

v) Production $V_1 \longrightarrow a$, $V_2 \longrightarrow b$, $A \longrightarrow V_1 A$ and. $B \longrightarrow V_2 B$ are already in CNF.

vi) Consider the production $S \longrightarrow V_1 A V_2 B$, we can rewrite as $S \longrightarrow V_3 V_4$, where, $V_3 \longrightarrow V_1 A$ and $V_4 \longrightarrow V_2 B$

Thus, the final CFG in CNF is

$S \longrightarrow V_3 V_4$

$V_3 \longrightarrow V_1 A$

$V_4 \longrightarrow V_2 B$

$V_1 \longrightarrow a$

$V_2 \longrightarrow b$

$A \longrightarrow V_1 A \mid a$

$B \longrightarrow V_2 B \mid b$

Imp

Q Consider an example

$S \longrightarrow AAC$

$A \longrightarrow aAb \mid \epsilon$

$C \longrightarrow aC \mid a$, convert the above grammar into CNF

Soln:- a) Remove ϵ - productions;

$A \longrightarrow \epsilon$, hence A is nullable

now,

$S \longrightarrow AAC$, $S \longrightarrow AAC$, $S \longrightarrow AAC$

$\longrightarrow \epsilon AC$ $\quad \longrightarrow A\epsilon C$ $\quad \longrightarrow \epsilon\epsilon C$

$\longrightarrow AC$ $\quad \longrightarrow AC$ $\quad \longrightarrow C$

also,

$A \longrightarrow aAb$

$\longrightarrow a\epsilon b$

$\longrightarrow ab$

$\therefore$ After eliminating ϵ-production, we have

$S \longrightarrow AAC/AC/C$

$A \longrightarrow aAb/ab$

$C \longrightarrow ac/a$

b) Remove unit production

Here, Unit production is $S \longrightarrow C$

Hence, Unit pair is (S, C)

$S \rightarrow C \rightarrow ac$

$S \rightarrow C \rightarrow a$

After removing unit production $S \longrightarrow C$, we have

$S \longrightarrow AAC/AC/ac/a$

$A \longrightarrow aAb/ab$

$C \longrightarrow ac/a$

c) Remove useless symbols.

Here, S, A, C are non terminals and all are generating. $\because$ they all provide terminal productions. also all these non-terminal symbols are reachable

Hence, we do not have any useless symbols.
Now, we can convert the grammar into CNF.

<u>Cycle 1:</u>

i) Productions $S \longrightarrow AC/a$ & $C \longrightarrow a$ are already in CNF.

ii) Consider the production $S \longrightarrow aC$, we can rewrite as, $S \longrightarrow V_1 C$, Where $V_1 \longrightarrow a$

also, $A \longrightarrow aAb$, we can rewrite as,
$A \longrightarrow V_1 A V_2$, where $V_2 \longrightarrow b$

also, $A \longrightarrow ab$ can be written as,
$A \longrightarrow V_1 V_2$

also, $C \longrightarrow aC$ can be rewritten as,
$C \longrightarrow V_1 C$

Thus

$S \longrightarrow AAC / AC | V_1 C | a$

$V_1 \longrightarrow a$

$A \longrightarrow V_1 A V_2 | V_1 V_2$

$V_2 \longrightarrow b$

$C \longrightarrow V_1 C | a$

<u>cycle 2:</u>

iii) for $S \longrightarrow AAC$, we can rewrite as
$S \longrightarrow V_3 C$, $V_3 \longrightarrow AA$

for $A \longrightarrow V_1 A V_2$, we can rewrite as
~~$A \longrightarrow V_4 V_2$, $V_4 \longrightarrow AV_2$~~

$A \longrightarrow V_4 V_2$, $V_4 \longrightarrow V_1 A$

Thus the final grammar in CNF will be as.

$S \rightarrow V_3C \mid AC \mid V_1C \mid a$

$V_3 \rightarrow AA$

$V_1 \rightarrow a$

$A \rightarrow V_4V_2 \mid V_1V_2$ $\qquad V_4 \rightarrow V_1A$

$V_2 \rightarrow b$

$C \rightarrow V_1C \mid a$

Greibach normal form :- (GNF)

⇒ A Grammar $G=(V,T,P,S)$ is said to be in GNF if its production rules are of the form

$$\alpha \rightarrow a\beta^*$$

Where, $a \in T$ & $\beta \in V$

ie.

non-terminal → exactly one terminal OR

non-terminal → one terminal followed by any no. of non terminals.

practice eg.

eg CNF Q Convert the following grammar into CNF

$S \rightarrow ASB \mid \epsilon$

$A \rightarrow aAS \mid a.$

$B \rightarrow SbS \mid A \mid bb$

Q $S \rightarrow ABCD$

$A \rightarrow aAb \mid \epsilon$

$C \rightarrow aC \mid a$

$D \rightarrow aDa \mid bDa \mid \epsilon$

$S \rightarrow aaaaS \mid aaaa$

Regular Grammar :-

⇒ A content free grammar $G = (V, T, P, S)$ is called a regular grammar if each production contains atmost one non-terminal on the right hand side of its production.

⇒ If the non terminal appears as the leftmost symbol on the right side of production, then the regular grammar is said to be ~~Right linear~~ ~~right~~ Left linear.

⇒ If the non-terminal appears as the rightmost symbol on the right side of production, then the regular grammar is said to be right linear.

- Left linear (Allowed types of production in left linear grammar)

$$A \longrightarrow BW$$
$$A \longrightarrow W$$
$$A \longrightarrow \epsilon$$

Where A & $B \in V$ i.e. A, B belongs to non-terminal	$W \in T$ i.e. W belongs to strings of terminals.

eg:-

$$S \longrightarrow Ca \mid Bb$$
$$C \longrightarrow Bba$$
$$B \longrightarrow Bab \mid b$$

- Allowed types of productions in <u>right linear grammar</u> are:-

$A \rightarrow wB$

$A \rightarrow w$

$A \rightarrow \epsilon$

Where A & B are nonterminals, w is a string of terminals.

<u>eg</u>:- $S \rightarrow 0A \mid 11B$

$A \rightarrow 01A \mid \epsilon$

$B \rightarrow 1110$

do Later

<u>Conversion of Right linear grammar to left linear grammar</u>:-

⇒ <u>steps</u>:-

① Represent the right linear grammar by a transition diagram.

② Interchange the position of start & final states.

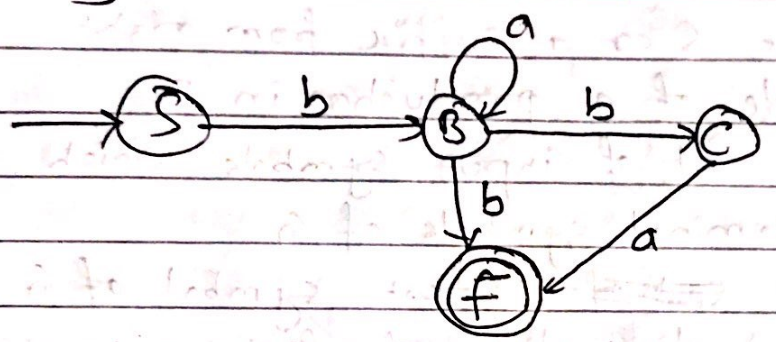
③ Reverse the directions of all the transitions keeping the positions of all intermediate states unchanged.

④ Re-write the grammar from transition graph in left linear fashion.

eg:- Convert following right linear grammar to left linear grammar.

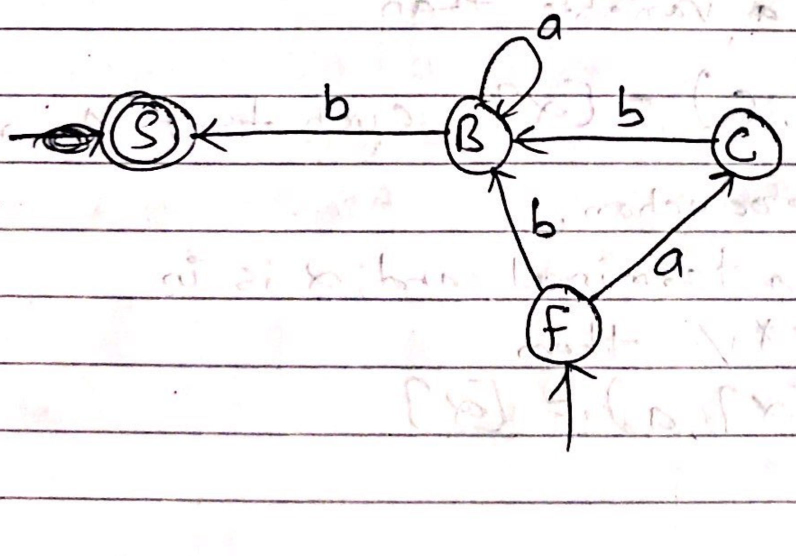
$S \longrightarrow bB$
$B \longrightarrow bC$
$B \longrightarrow aB$
$C \longrightarrow a$
$B \longrightarrow b$ } to final state

Soln:- Representing the right linear grammar by transition diagram as,



eg baaaba

- Interchanging the position of start & final states, also, reversing the directions of all the transitions.



- rewrite the grammar as,

~~$F \longrightarrow aC \mid bB$~~
~~$B \longrightarrow aB \mid bS$~~

$F \longrightarrow Ca \mid Bb$
$B \longrightarrow Ba \mid Sb \mid b$ To final state
$C \longrightarrow Bb$
~~$S \longrightarrow Sa$~~
$S \longrightarrow \epsilon \mid a \mid b$

RG to NFA :-

Another way:-

- If the regular grammar is of the form in which all the productions are in the form as;

$A \to aB$ or $A \to a$,

where

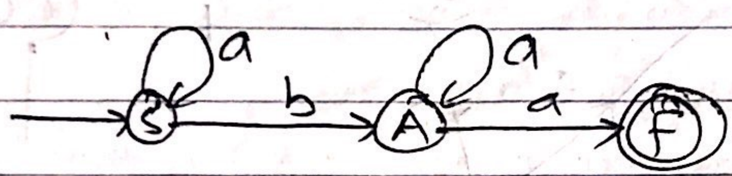
$A, B \in V$ and $a \in T$

eg

$S \to aS \mid bA$

$A \to a \mid aA$

{ if n variable then n+1 states }

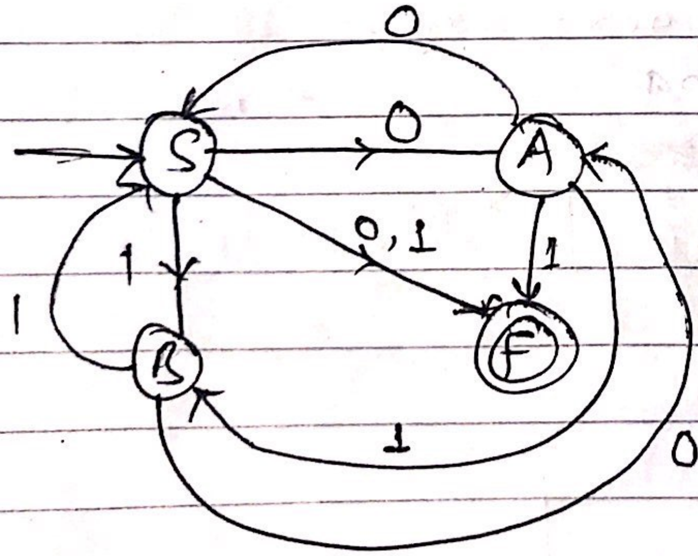


eg

$S \to 0A \mid 1B \mid 0 \mid 1$

$A \to 0S \mid 1B \mid 1$

$B \to 0A \mid 1S$



eg

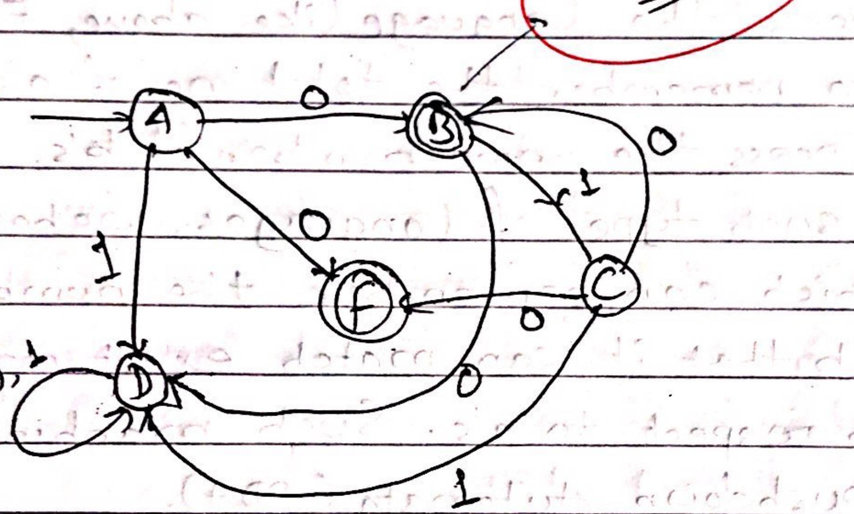
$A \to 0B \mid 1D \mid 0$

$B \to 0D \mid 1C \mid \epsilon$

$C \to 0B \mid 1D \mid 0$

$D \to 0D \mid 1D$

$\therefore B \to \epsilon$



eg

$A \to 0B \mid E \mid 1C$

$B \to 0A \mid F \mid \epsilon$

$C \to 0C \mid 1A \mid 0$

$E \to 0C \mid 1A \mid 1$

$F \to 0A \mid 1B \mid \epsilon$