

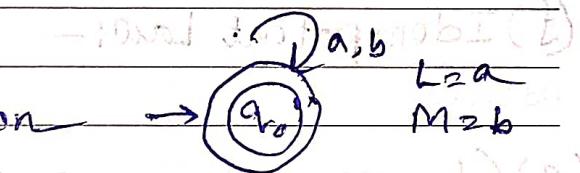
Properties of Regular Expression:-

Algebraic Laws for Regular Expr

(a) Commutative — $xy = yx$ [+]

$L \& M$ regular expression

$$S = L + M = M + L$$



$$ab \neq ba$$

(b) Associativity — a) $(L + M) + N = L + (M + N)$

$$b) (LM)N = L(MN)$$

(c) Distributive — $L(M+N) = LM + LN$ (Left Distributive Law)

F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	OCT 2021

$$(M+N)L = ML + NL \text{ (Right Distributive Law)}$$

02

OCTOBER
SATURDAY | 2021૧૧-૧૨ આશ્વિન કૃષ્ણ શનિ-રવિ ૨૦૭૮
૧૫-૧૬ આશ્વિન શનિ-રવિ ૧૪૨૮
એકાદશી રા ૧/૯૦-દ્વાદશી રા ૧/૯૫
Hizri-24-25 Safar 1443Theorem (Based on Distributive law):—

If L, M, N are regular language then $L(M+N) = LM + LN$

$$\begin{array}{c|c|c} \text{So } w = x \cdot y & \\ \text{If } w \in LM + LN & \text{So, } x \in L \text{ and } y \in (M+N) & w = L \cdot (M+N) \\ \downarrow & \downarrow & \Rightarrow y \in M \text{ or } y \in N \\ xy \in LM \text{ or } xy \in LN & & w = xy \\ \Rightarrow x \in L \text{ and } y \in M \text{ or } x \in L \text{ and } y \in N & & \end{array}$$

$w = LM + LN, w = xy$ contains L
 $xy \in LM$ or $xy \in LN$ contains M or N
 $\Rightarrow x \in L$ and $y \in M$ or $x \in L$ and $y \in N$ others N

(d) Identity: —

$$\emptyset + S = S$$

$$\emptyset + L = L + \emptyset = L$$

$$\therefore L = L + L = L$$

\emptyset is identity for union

(e) Annihilators: —

$$\emptyset \cdot L = L \cdot \emptyset = \emptyset$$

$$L \cdot \emptyset = \emptyset \cdot L = \emptyset$$

(f) Idempotent Law: —

03 Sunday

(g) Closure: —

$$(L^*)^* = L^* \quad \emptyset^* = \epsilon \cdot M \cdot \epsilon^* = \epsilon$$

$$L^+ = LL^*$$

$$L? = \epsilon + L$$

Language

Closure Properties of Regular Expression:—

a) union — The union of 2 Regular Language is regular.

OCT.	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	F	S
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22

(and annihilator) $L + LM = L(L + M)$

१३ आश्विन कृष्ण सोमवार २०७८
 १७ आश्विन सोमवार १४२८
 अद्योतनी रा १/३०/७१
 Hizri-26 Safar 1443

OCTOBER
2021 MONDAY

04

b) Intersection

(*) Infinite Union is not a closure property of R.L

c) Complement

d) Difference

e) Reversal

f) Closure

g) Concatenation

h) Homomorphism - (substitution of strings for a symbol of a regular language)

0101

$h(0) = aa$

0101

$h(1) = b$

i) Inverse Homomorphism

$\rightarrow h^{-1}(aa) = 0$

Th. If L & M are regular languages then so is LUM.

LUM

$LUM = L(R) \cup L(S)$

$L = L(R)$
 $M = M(S)$

$= L(R \cup S)$

Th - If L is Reg. Language then the complement \bar{L}^0 is also a Regular Language.

$\bar{L} = \Sigma^* - L$

$A' = (Q, \Sigma, \delta_0, S, Q-F)$

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	F	S	S	M	T	W	NOV							
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

ৱেপাৰ মুসু

৭৪ আশিন কৃষ্ণ মন্দির বার ২০০৮

১৮ আশিন মন্দির বার ১৪২৮

চতুর্দশী রা ৬/৩২/৩৭

Hijri-27 Safar 1443

০৫

OCTOBER
TUESDAY
2021

If L & M is Regular Language then $L \cap M$ is R.L.

Proof :- $L \cap M = \overline{\Sigma \cup M}$

$$A_M = (Q_M, \Sigma, \delta_M, q_{M_0}, F_M)$$

$$A_L = (Q_L, \Sigma, \delta_L, q_{L_0}, F_L)$$

The finite automata for $L \cap M$ is A

P, q are 2 states where A gives $p \in A_M$ and $q \in A_L$

$$\delta_L(p, a) = s$$

$$\delta_M(q, a) = t$$

$s \in A_L$ and $t \in A_M$

$$\delta((p, q), a) = (s, t) = (\delta_L(p, a), \delta_M(q, a)) \quad (i)$$

$$A = (Q_L \times Q_M, \Sigma, (q_{L_0}, q_{M_0}), \delta, F_L \times F_M)$$

Now we want to prove that A is R.L.

If $L - M = L \cap \overline{M}$ [Closure Under Difference]

If L is R.L then the reversal of L (L^R) is also R.L.

If L is R.L then Homomorphism of L is also R.L.

If L is R.L then inverse-Homomorphism of L is also R.L.

OCT.	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

OCTOBER
 2021 WEDNESDAY

06

How to prove a Language is not Regular?

Pumping Lemma for Regular Language: - ~~সুন্দর লেম্মা~~

$L = \{0^n 1^n : n \geq 0\}$ not a regular language

Lemma →

Let L be a R.L. then there exists a constant n (which depends on L) such that for every string w in L such that $|w| \geq n$, we can break w into 3 strings $w = xyz$ such that - (i) $y \neq \epsilon$ / (ii) $|xy| \leq n$ / (iii) for all $k \geq 0$, the string $xy^k z$ is also in L

$L = L(A)$

L is a regular language for which there is a finite automata A

$w = a_0 a_1 \dots a_m$ (where $m \geq n$) and each a_i is the input symbol

For $i = (0, 1, \dots, n)$ we can define the state p_i as

$$\hat{g}(q_0, (a_0 a_1 a_2 \dots a_i)) = p_i$$

and q_0 is the start state of A

From $i \dots j$, we have $\hat{g}(p_i, (a_i \dots a_j)) = p_j$

such that $0 \leq i \leq j \leq n$

$$\rightarrow \textcircled{p}_0 \rightarrow \textcircled{p}_i \rightarrow \dots \rightarrow \textcircled{p}_j$$

$y = a_{i+1} \dots a_j$

$$x = a_0 \dots a_i$$

$$z = a_{j+1} \dots a_n$$

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	N													
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

07

OCTOBER
THURSDAY | 2021

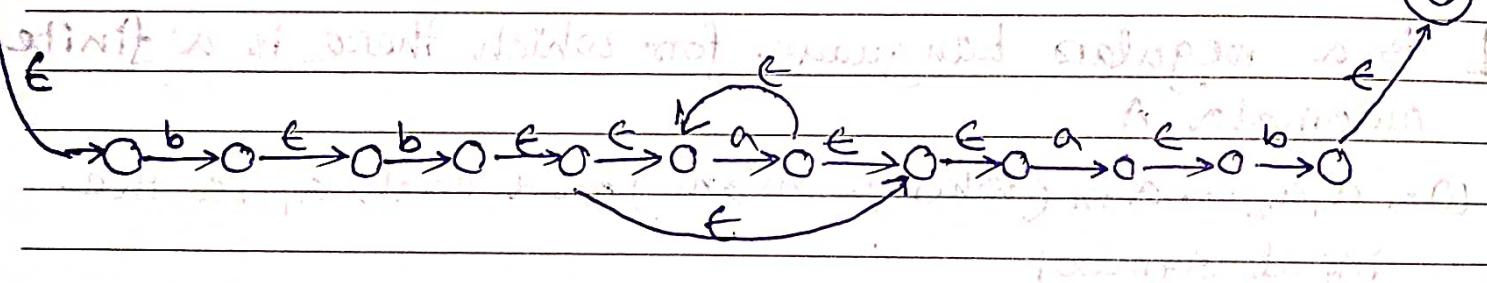
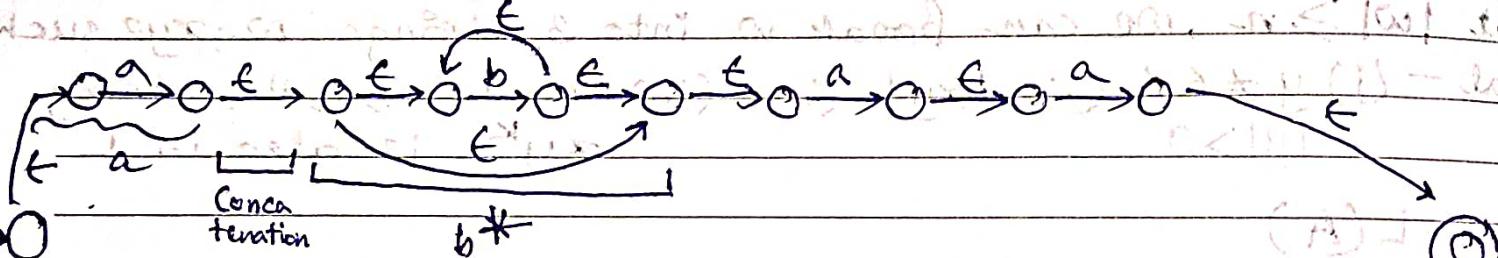
৭ আরিবন শুক্রবার ২০৮৫
 ২০ আশ্বিন বৃহস্পতিবার ১৪২৮
 প্রতিপদা ষ ৩/২৫/৩৮
 Hizri-29 Safar 1443

Automata Exam Solution

$$Q. b) L = (a+b)^* b (a+bb)^*$$

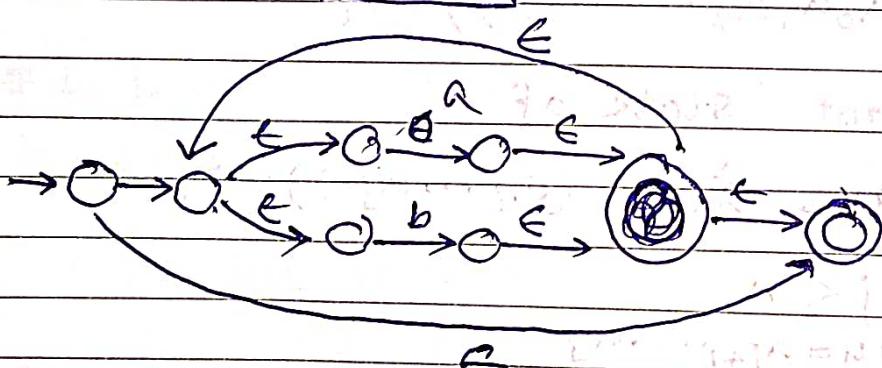
Thomson Construction :-

$$L = (ab^*aa) + (bb a^* ab)$$



Ans 14 STATE WITH INITIAL AND FINAL STATE (1, 2, 3) = 1 + 0.1

$$(a+b)^*$$



OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S							
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

12

OCTOBER
TUESDAY
2021

Q/ Prove $a^n b^n$ regular Language are not. $[a^n b^n]$

Ans:- not regular $\omega = xyz$

Assume A is regular

Pumping Length = p

$$S = a^p b^p$$

$$x = a^{p-1} \quad y = \underline{a} \quad z = b^p$$

$$y = a \quad z = b^p$$

$$xy^2z = a^{p-1} a^2 b^p$$

$$= a^{p+1} b^p \notin A$$

\therefore not regular

$$|xyz| \leq n$$

$$xyz^k \text{ where } k \geq 0$$

Q/

$$\omega = \underline{a}^{2n}, n \geq 0.$$

$$\omega = xyz$$

$$y \neq \epsilon$$

$$x = \epsilon$$

$$y = 00$$

$$z = \epsilon$$

$$xy^k z$$

$$k=0 \quad \epsilon$$

$$k=1 \quad 00$$

$$k=2 \quad 0000$$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Regular Grammars: - (Type-3)

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

$$V = \{A, B\} \quad T = \{a\} \quad S = \{A\}$$

$$P \Rightarrow$$

$$A \rightarrow aB$$

$$B \rightarrow a$$

$$A \Rightarrow a$$

$$A \rightarrow \epsilon$$

$$\underline{A} \rightarrow \underline{AB} \mid \underline{B}$$

Left Linear Grammars

$$\underline{A} \rightarrow \underline{BA} \mid B$$

Right Linear Grammars

Context Free Grammars (Type-2): -

→ It's have some recursive nature.

→ Parsing into DTD → LATEX

Document type definition

(1) There is a finite set of symbols that form the strings of the language being defined. We can call this Alphabet the Terminals / Terminal Symbols.

(2) There is a finite set of variable also called sometimes not-terminals or syntactic categories. Each variable represent a ^{Language} symbol that is a set of strings.

(3) One of the variables represents the language being defined. It is called the start symbols. Other variable represents auxiliary classes of strings that are used to help define the language of ~~the~~ start symbol.

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	NOV						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

14

OCTOBER
THURSDAY 2021

④ There is finite set of productions or production rules that represent the recursive defⁿ of a language. Each production consists of -

(a) A variable that is being (partially) defined by the production is called the head of the production.

(b) The production symbol arrows (\rightarrow)

(c) A string of 0 or more terminals and variables. This string is called the body of the production.

From

The body of the production there is one way to form strings in the language of the variable head. In doing so we leave the terminals unchanged and substitute for each variable of the body of any string that is known to be in the language of that variable.

Recursive Inference

Leftmost derivation \rightarrow

$$S \xrightarrow{Lm} AB \xrightarrow{Lm} a \underline{A} B \xrightarrow{Lm} a a \underline{B} \xrightarrow{Lm} a a b$$

$$S \rightarrow AB$$

$$A \rightarrow aA | a$$

$$B \rightarrow aB | b$$

$$S \xrightarrow{* Lm} aab$$

aab can be derived from S in 0 or more steps.

Rightmost derivation \rightarrow

$$S \Rightarrow A \underline{B} \Rightarrow \underline{A} b \Rightarrow a A b \Rightarrow a a b$$

If Context-Free Language (CFL): -

If $L(V, T, S, P)$ is a CFG the Language of L is $L(L)$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	F	S	S
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Today's topic is to express with English given

OCTOBER
2021 FRIDAY

15

is the set of terminal strings that have derivation from the start symbol.

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

If a Language L of some context free grammars then L is called CFL.

~~Parse Tree~~

$$E \rightarrow I$$

$$(a101 + b1)^* (a1 + b)$$

$$E \rightarrow E + E$$

most Left Derivation

$$E \rightarrow E * E$$

$$E \rightarrow (E)$$

$$I \rightarrow a$$

$$I \rightarrow b$$

$$I \rightarrow I_a$$

$$(I01 + E)^* E \xleftarrow{lm} (I1 + E)^* E \xleftarrow{lm} (I + I)^* E$$

$$I \rightarrow I_b$$

$$I \rightarrow I_0$$

$$I \rightarrow I_1$$

$$(a101 + b1)^* (E + E) \xleftarrow{lm} (a101 + b1)^* (E) \xleftarrow{lm} (a101 + b1)^* E$$

$$(a101 + b1)^* (I_1 + E) \xrightarrow{lm} (a101 + b1)^* (a1 + E) \xrightarrow{lm} (a101 + b1)^* (a1 + I) \xleftarrow{lm} (a101 + b1)^* (a1 + b)$$

H.W → Right most Derivation

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	NOV						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

16

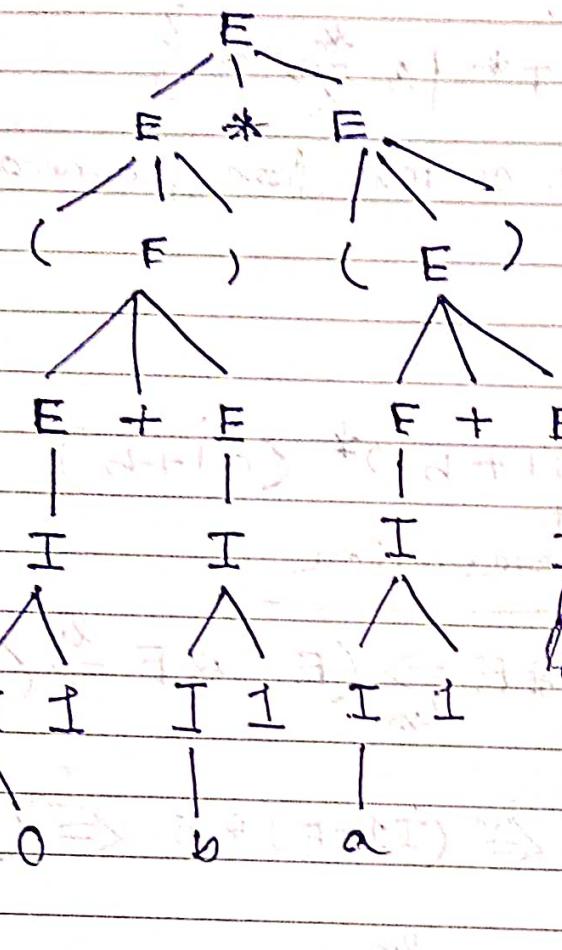
OCTOBER

SATURDAY | 2021

Parse Tree

୧୧-୧୨ ଆଶିନ ଶୁଵଳ ଶନି-୧ବି ୨୦୯୮
୨୯-୩୦ ଆଶିନ ଶନି-୨ବି ୧୪୨୮
ଏକାଦଶୀ ତା ୧/୧୩ ଦ୍ୟାଦଶୀ ତା ୬/୩୩
Hizrl-9-10 Rabl-ul-awwal 1443

Parse tree for $(a101+b1) * (ab\ a1+b)$



[Root is the start
Symbol of the grammar]

17 Sunday

a

Grammars for which distinct parse tree cannot be obtained are called ambiguous grammars.

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S							
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Kidney Disease \leftarrow Hypertension

OCTOBER
2021 MONDAY

18

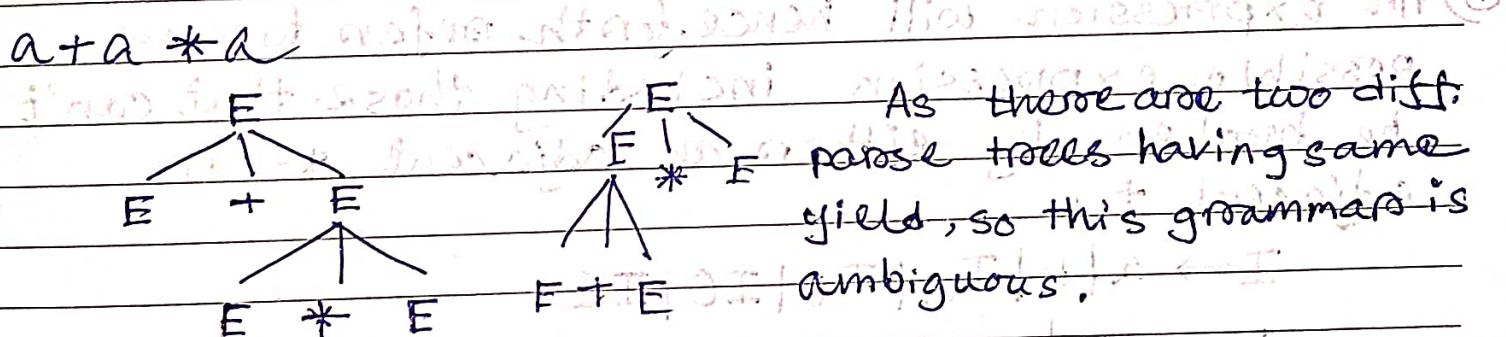
From Inferences to Trees:-

Defn - Let $G = (V, T, S, P)$ be a CFC if the recursive inference procedure tells us that terminal string w is in the language of variable A , then there is a parse tree with root A and yield w .

Ambiguous Grammars :-

$$E \rightarrow E+E \quad | \quad E * E$$

$$E \rightarrow a$$



Rules to remove ambiguity from the grammar:-

- 1) A factor is an expression that cannot be broken apart by any adjacent operators $*$ or $+$. The only factors in our expression language are —

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	NOV						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

19

OCTOBER
TUESDAY | 2021

(a) Identifiers — It is not possible to separate the letters of an identifier by attaching an operator.

(b) Any parenthesized expression; no matter what appears inside the $(\)$ to present what is inside from becoming the operand of any of the operators outside the $(\)$.

2) A term is an expression that cannot be broken by + operator.

$$\text{Eg. } a+a*a \Rightarrow a+(a*a)$$

$$a*a*a \Rightarrow (a*a)*a$$

3) An expression will henceforth refer to any possible expression including those that can't be broken by either an adjacent $*$ or adjacent $+$.

$$I \rightarrow a|b|I_a|I_b|I_0|I_1$$

$$F \rightarrow I|CE$$

$$T \rightarrow F|T*F$$

$$E \rightarrow (T)|E+I$$

The form a grammar $G = (V, T, S, P)$ and string w in T^* ; w has two distinct parse tree iff w has two distinct left most derivations from S .

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S							
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

OCTOBER
 2021 WEDNESDAY

20

Inherent Ambiguity :-

$$L = \{a^n b^m c^m d^n | n \geq 1, m \geq 1\} \cup \{a^n b^m c^m d^n | n \geq 1, m \geq 1\}$$

$$W = aabbccdd$$

$$S \rightarrow AB|C$$

$$A \rightarrow aAb|ab$$

$$B \rightarrow cBd|cd$$

$$C \rightarrow aCd|add$$

$$D \rightarrow bDc|bc$$

$$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aabbB$$

$$A \leftarrow a$$

$$\Downarrow$$

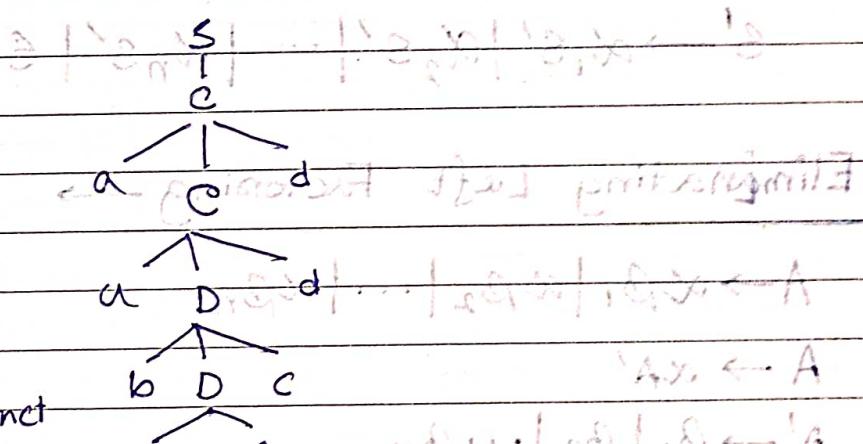
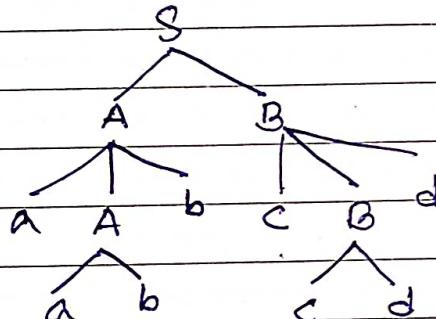
$$aabbBd$$

$$\Downarrow$$

$$aabbacd$$

$$S \Rightarrow C \Rightarrow aCd \Rightarrow aadd$$

$$S \Rightarrow A \Rightarrow aAb \Rightarrow aabb$$



As there are two distinct parse trees and this ambiguity can't be removed, so this grammar has inherent ambiguity.

Left-recursive Grammar:-

$$S \rightarrow S\alpha . \beta$$

~~It is a grammar in which a non-terminal appears on the left side of its own production rule.~~

We can rewrite this as

$$\boxed{\begin{aligned} S &\rightarrow \beta S' \\ S' &\rightarrow \alpha S' | \epsilon \end{aligned}}$$

M	T	W	T	F	S	S	S	M	T	W	T	F	S	S	M	T	W	.NOV
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	2021

21

OCTOBER
THURSDAY | 2021

१ कार्तिक कृष्ण गुरुवार २०७८
 ३ कार्तिक बृहस्पतिवार १४२८
 प्रतिपद रा ८/५६/८
 Hizri-14 Rabi-ul-awwal 1443

$$\text{Eq. } E \rightarrow E(T) \overline{T} \xrightarrow{\infty} \beta$$

$$E \rightarrow TE'$$

$$E' \xrightarrow{\infty} T E' | \epsilon$$

$$\begin{aligned} S &\rightarrow A \\ A &\rightarrow Sd/b \end{aligned} \quad \left. \begin{array}{l} \text{Indirectly} \\ \text{LR} \end{array} \right\}$$

$$S \xrightarrow{*} Sd$$

$$S \rightarrow A$$

$$S \rightarrow Sd/b$$

$$S \rightarrow bS'$$

$$S' \rightarrow dS' | \epsilon$$

$$S \rightarrow S\alpha_1 | S\alpha_2 | \dots | S\alpha_n | \beta_1 | \beta_2 | \dots | \beta_n$$

$$S \rightarrow \beta_1 | \beta_2 | \dots | \beta_n$$

$$S' \rightarrow \alpha_1 S' | \alpha_2 S' | \dots | \alpha_n S' | \epsilon$$

Eliminating Left Factoring

$$A \rightarrow \alpha \beta_1 | \alpha \beta_2 | \dots | \alpha \beta_n$$

$$A \rightarrow \alpha A'$$

$$A' \rightarrow \beta_1 | \beta_2 | \dots | \beta_n$$

Linear Grammar :-

$$A \rightarrow B$$

Left Linear grammar (when non terminals are at the extreme left)

$$S \rightarrow Sa$$

$$S \rightarrow Ab$$

$$A \rightarrow a$$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S								
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

২ কার্তিক কৃষ্ণ শুক্লবার ২০৭৮

৪ কার্তিক শুক্লবার ১৪২৮

বিত্তীয়া রা ১০/৩৮/২০

Hizri-15 Rabi-ul-awwal 1443

Right Linear Grammar:-

$$S \rightarrow aS$$

$$S \rightarrow bA$$

$$A \rightarrow a$$

23

OCTOBER | SATURDAY 2021

३-४ कार्तिक कृष्ण शनि-रवि २०७८
 ५-६ कार्तिक शनि-रवि १४२८
 ढ़तीया रा १२/३८-चतुर्थी रा २/४६
 Hizri-16-17 Rabi-ul-awwal 1443

Simplification of Grammar:

1. Eliminate null production
2. Eliminate unit production
3. " Useless production

1.

$$S \rightarrow ABaC$$

$$A \rightarrow BC$$

$$B \rightarrow b|E$$

$$C \rightarrow D|E$$

$$D \rightarrow d$$

$$B \rightarrow E \quad \text{Nullable}$$

$$C \rightarrow E$$

CNF ANF

$$S \rightarrow A B a C \quad | \quad S \rightarrow A B a C$$

$$S \rightarrow B C a C$$

$$S \rightarrow a$$

$$A \rightarrow B C \Rightarrow A \rightarrow E$$

$$\begin{matrix} | \\ E \\ | \end{matrix}$$

$$A \Rightarrow B C \Rightarrow C \Rightarrow E$$

$$A \rightarrow E$$

24 Sunday

$$S \rightarrow ABaC | BaC | AaC | ABa |$$

$$aC | Aa | Ba | a$$

$$A \rightarrow BC | C | B$$

$$B \rightarrow b$$

$$C \rightarrow D$$

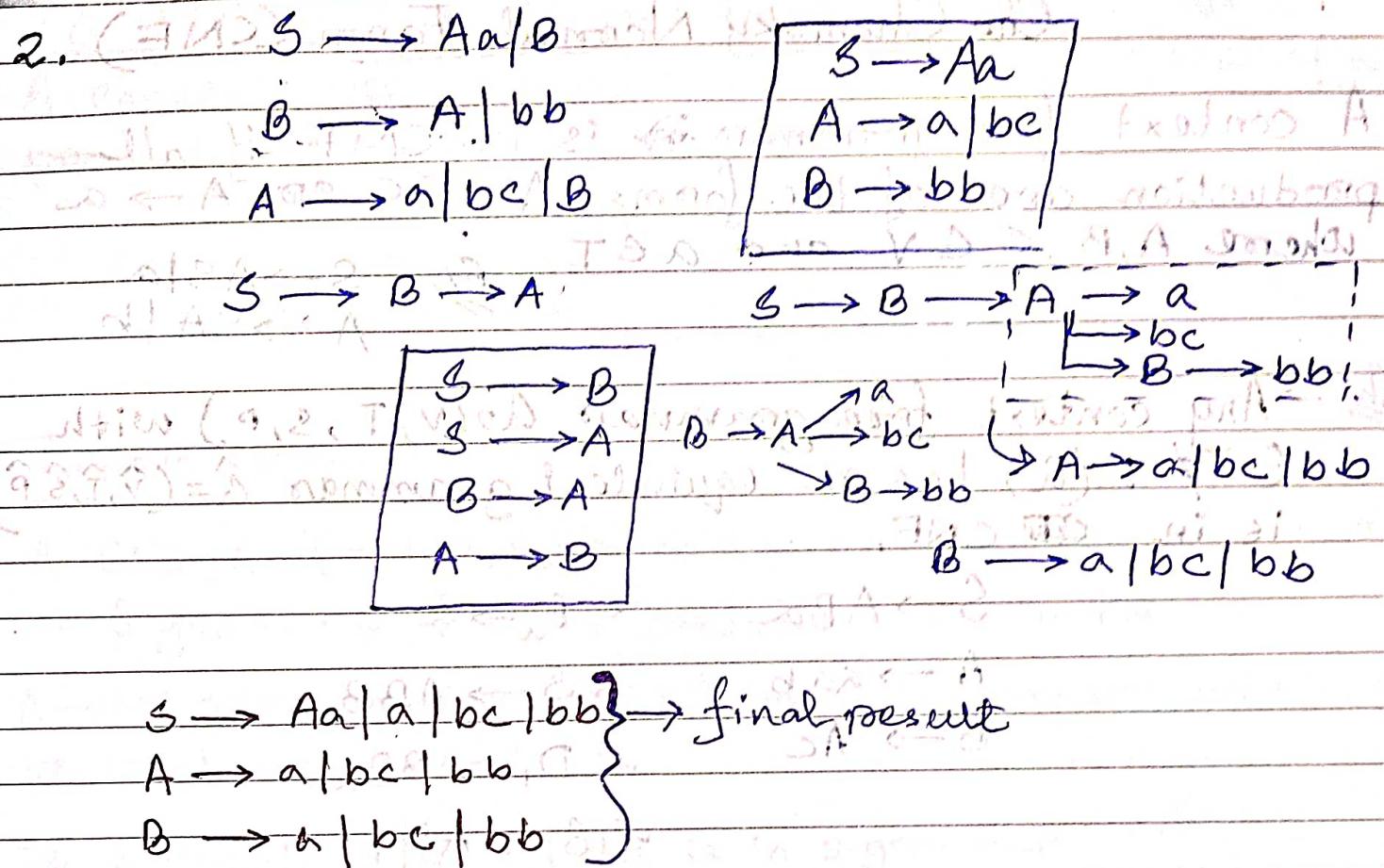
$$D \rightarrow d$$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S								
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

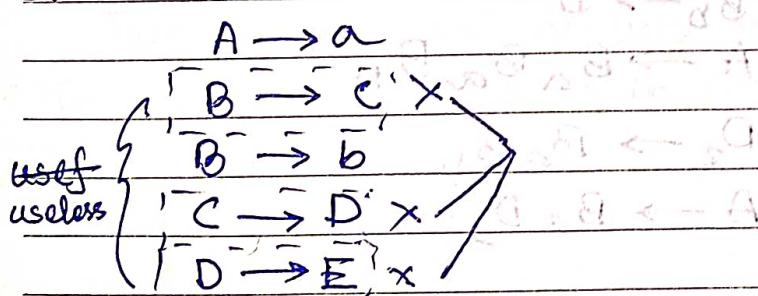
৫ কার্তিক কৃষ্ণ সোমবার ২০৭৮
 ৭ কার্তিক সোমবার ১৪২৮
 পঞ্চমী শে ৮/১০/৬
 Hizri-18 Rabi-ul-awwal 1443

OCTOBER
2021 MONDAY

25



3. $S \rightarrow AB$



$SA \leftarrow b \times$

$\rightarrow C \rightarrow \bar{b}$

$SA \leftarrow B$

(9, 2, T, V) = 3

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	.NOV						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

৩২৫ * ৩ = T

Chomsky Normal Form (CNF)

A context free grammar is in CNF if all production are of the forms $A \rightarrow BC$ or $A \rightarrow a$ where $A, B, C \in V$ and $a \in T$

$$\text{Ex} - S \rightarrow AS1a \\ A \rightarrow SA1b$$

\Rightarrow Any context free grammar $G_2(V, T, S, P)$ with $E \notin L(G)$ has an equivalent grammar $\hat{G} = (\hat{V}, \hat{T}, \hat{S}, \hat{P})$ is in CNF.

$$S \rightarrow ABA \quad \checkmark B_a \rightarrow a$$

$$A \rightarrow aab \quad \times S \rightarrow ABBB_a$$

$$B \rightarrow AC$$

$$\checkmark D_1 \rightarrow BB_a$$

$$\checkmark S \rightarrow AD_1$$

$$\times A \rightarrow aab$$

$$B_b \rightarrow b$$

$$\times A \rightarrow B_a B_a B_b$$

$$D_2 \rightarrow B_a B_b$$

$$A \rightarrow B_a D_2$$

$$\times B \rightarrow Ac$$

$$B_c \rightarrow c$$

$$B \rightarrow AB_c$$

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

$$V = \{S, A, B, B_a, B_b, B_c, D_1, D_2\}$$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S								
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

$$T = \{a, b, c\}$$

$$S = \{S\}$$

६ कार्तिक कृष्ण बुधवार २०७८
९ कार्तिक बुधवार १४२८
बस्ती घ ६/८१/८१
Hizri-20 Rabi-ul-awwal 1443

OCTOBER
2021 | WEDNESDAY

27

Anpan
Basu

CNF (Chomsky Normal Form)

A context free grammar is said to be CNF if all productions have the form

$$A \rightarrow a \text{ or } X$$

where $a \in T$

$$X \in V^*$$

Simple Grammars or S-grammars:-

A CFG, $G = (V, T, S, P)$ is said to be a simple grammar or S-grammar if all its productions are in the form

$A \rightarrow aX$ where $a \in T$ and $X \in V^*$ and $A \in V$ and any pair (A, a) occurs at most once in P .

Eg. $S \rightarrow aS \mid bSS \mid c$] this is in s-grammar

Eg. $S \rightarrow \underline{a}S \mid b\underline{SS} \mid \underline{a}SS \mid c$

This pair $S \rightarrow a$ occurs more than once. So this is not in s-grammar. But it is CNF.

$$\begin{array}{c} AAS \leftarrow Aa \leftarrow AAa \leftarrow A \\ a \leftarrow Aa \quad a \end{array}$$

$$\{ S \leftarrow a \quad \} S \mid Aa \mid AAS \leftarrow A$$

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

• NOV

2021

28

OCTOBER

THURSDAY | 2021

ੴ ਕਾਰਿਕ ਕ੃ਣਾ ਗੁਰੂਵਾਰ ੨੦੭੮
 ੧੦ ਕਾਰਿਕ ਵਹਿਸਤਿਵਾਰ ੧੪੨੮
 ਸ਼ੁਭਮੀ ਘ ੮/੧੦/੫੯
 Hizri-21 Rabi-ul-awwal 1443

Q

 $S \rightarrow AB$

(Create Janmohi (condition)) THIS

 $A \rightarrow aA \mid bB \mid b$ $A \rightarrow aX$ $B \rightarrow b$ $a \in T$ $S \rightarrow AB$ $A \in V$ $S \rightarrow AAB \mid bBB \mid bB$ $X \in V^*$

be the grammar after continuing to CNF is

 $S \rightarrow aAB \mid bBB \mid bB$ ((S, b) and (A, b) occurs twice so it's not S-grammar) $A \rightarrow aA \mid bB \mid b$ b $B \rightarrow b$ (S, a) also into two $V \ni A$ bao \rightarrow ~~5~~ bao Tba \rightarrow ~~5~~ bao $\times \leftarrow A$ $S \rightarrow abSb \mid aa$ $A \rightarrow a$ $B \rightarrow b$ $S \rightarrow aBSB \mid aA$

This is in CNF form.

 $A \rightarrow ab$ $B \rightarrow b$

Q

 $S \rightarrow AB$ $A \rightarrow aAA \mid \epsilon$ $B \rightarrow bBB \mid \epsilon$ Eliminate ϵ -production from this.
$$\begin{array}{c} A \xrightarrow{\epsilon} AAA \xrightarrow{\epsilon} aA \xrightarrow{\epsilon} aaa \\ \downarrow \quad \downarrow \quad \downarrow \\ \epsilon \quad aA \xrightarrow{\epsilon} a \end{array}$$

OCT	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S														
2021	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

$$\begin{array}{c} A \xrightarrow{\epsilon} AAA \{ aA \mid a \} \\ S \rightarrow B \end{array}$$

29

$$B \rightarrow bBB | bB | b \quad S \rightarrow A \\ S \rightarrow AB | A | B \quad \text{Ans.}$$

Removing non-terminals

$$S \rightarrow AB | aAA | aA | a | bBB | bB | b$$

$$A \rightarrow aAA | aA | a$$

$$B \rightarrow bBB | bB | b$$

Removing useless productions,

$$S \rightarrow AB | aA | a | bB | b$$

$$A \rightarrow aA | a$$

$$B \rightarrow bB | b$$

Eg: Convert the following grammars in CNF.

a) $S \rightarrow ABA$

A $\rightarrow aab$

B $\rightarrow AC$

b) $S \rightarrow aSa | sSa | a$

c) $S \rightarrow a | b | cSS$

Convert the following grammars in GNF.

a) $S \rightarrow aSb | ab$

b) $S \rightarrow ABb | a$

A $\rightarrow aaaA | B$

B $\rightarrow bAb$

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	N													
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021

44th Wk • 303-062

30

OCTOBER

SATURDAY

2021

९-१० कार्तिक कृष्ण शनि-रवि २०७८

१२-१३ कार्तिक शनि-रवि १४२८

नवमी घ ९/४८-दशमी घ ९/५०

Hizri-23-24 Rabi-ul-awwal 1443

$$a) S \rightarrow ABa$$

$$A \rightarrow aab$$

$$B \rightarrow Ac$$

$$S \rightarrow abAB$$

$$A \rightarrow bAB | E$$

$$B \rightarrow BAa | A | E$$

$$A \rightarrow E$$

$$S \rightarrow abAB | abB$$

$$A \rightarrow bB bAB | bB$$

$$B \rightarrow BAa | A | Ba | E$$

$$B \rightarrow E$$

$$S \rightarrow abAB | abB | ab | aba$$

$$A \rightarrow bAB | bB | bA | b$$

$$B \rightarrow BAa | A | Ba | Aa | a$$

१७ कार्तिक कृष्ण सोमवार २०७८
१८ कार्तिक सोमवार १४२८
एकादशी घ ९/२२/२९
Hizri-25 Rabi-ul-awwal 1443

45th Wk • 305-060

NOVEMBER
2021 MONDAY

01

module name (input a, input b, output c);

begin
assign c = a & b;
end
end module

$$(A, S, P, G, T, Z, D) = ?$$

Test

module testabc

a = , x

b = , y
c = , z

initial

x=1;
y=0;

#100;

end module;

module name (a(x); b(y); c(z));

input x = A - B

input y = C - D

2) Y = A + B and Z = X - Y = X - A - B
so that Z = X - A - B

that is value of Z is equal to X - A - B

M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	.NOV						
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	• 2021