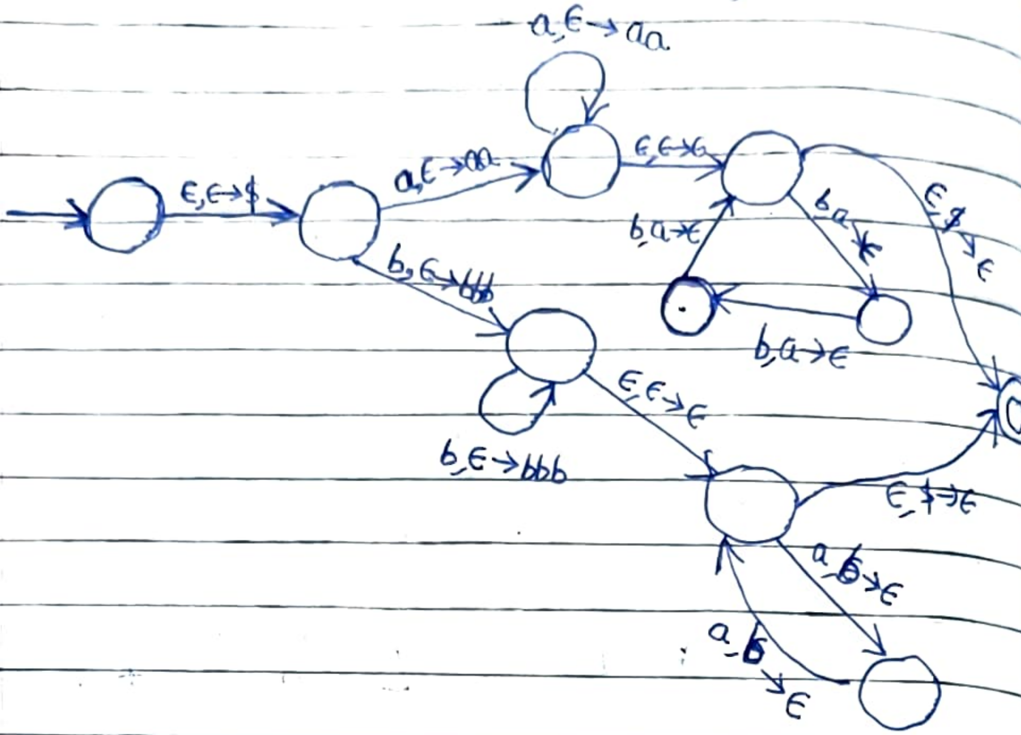
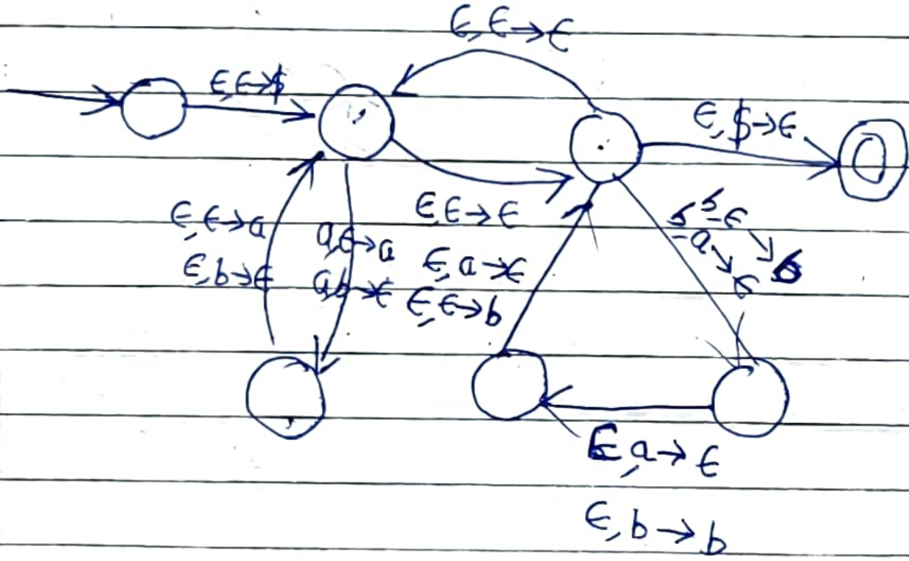
$L = \{ w \in (a+b)^* \mid 2|w_a| = 3|w_b| \}$





ab aab

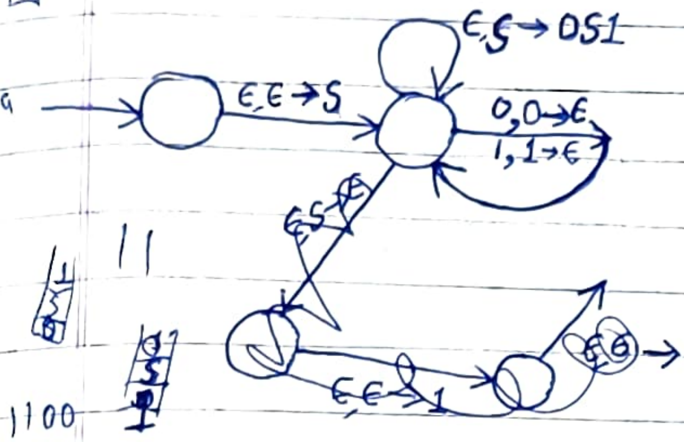
46

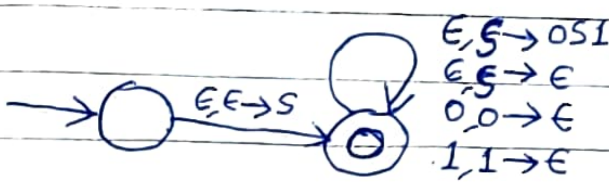
15

Equivalance b/w CFG & PDA

$L(G) = L(M)$

$S \to 0S1 \mid \epsilon$





① For every CFG we construct a PDA as follow

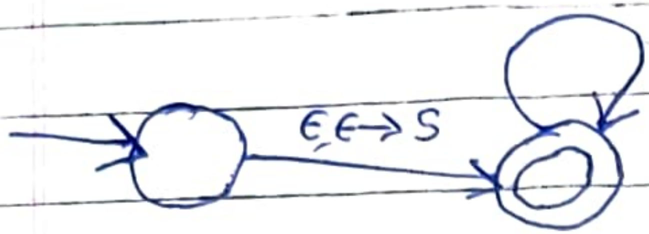
(a) Insert the start symbol to stack on ϵ

(b) Create transitions $A \to w$ on second state
$(\epsilon, A \to w)$

(c) $\forall\ a \in \Sigma$ add transition $a, a \to \epsilon$

Date:
Page:

$S \to 0S1S \mid 1S0S \mid \epsilon$



$\epsilon, S \to 0S1S$

$\epsilon, S \to 1S1S$

$\epsilon, S \to \epsilon$

$1, 1 \to \epsilon$

$0, 0 \to \epsilon$

0110

$S \to \boxed{aaa}Sbb$

~~aSaa~~ S

$\frac{6!}{3!\,2!}$ $\frac{720}{6 \times 2} = 60$

Saaabb

aaabbS

23/2/23

CS203

PDA to CFG conversion
Deterministic PDA
Chomskey Normal Form of CFG
Pumping lemma & Non-context free language.

A context free grammer is said to be in Chomskey normal form if every production rule is of the form $A \to BC$ or $A \to a$

$B, C \in N \quad a \in T$

Only $S \to \epsilon$ is allowed & S will not appear in RHS of any rule.

String of length n $\Rightarrow$ 2n steps.

Every context free grammer ~~there~~ is an equivalent grammer with chomskey normal form.

Strategy

1. Introduce a new start symbol.
2. Get rid of all ϵ- transitition,
3. Get rid of the production where RHS is one variable
4. Convert long rules into short rules by introducing new termi'l

$S \rightarrow aXbX \mid \cancel{aY} \mid \cancel{bY}$
$Y \rightarrow X \mid c$
$X \rightarrow aY \mid bY \mid \epsilon$

$S \rightarrow aXbX \mid ab \mid aXb \mid abX$
$Y \rightarrow X \mid c$
$X \rightarrow aY \mid bY \mid a \mid b$

$Y \rightarrow aY \mid bY \mid a \mid b \mid c$

$S \rightarrow EF \mid GF \mid EH \mid GI$
$E \rightarrow GX$
$F \rightarrow b$
$G \rightarrow a$
$H \rightarrow FY$
$I \rightarrow FX$

$X \rightarrow GY \mid FY \mid a \mid b$
$Y \rightarrow GY \mid FY \mid a \mid b \mid c$

(E) $S \to aSb \mid \epsilon$

$S_0 \to \epsilon \mid S$
$S \to aSb \mid ab$

$S_0 \to \epsilon \mid AC \mid AB$
$S \to AC \mid AB$
$A \to a$
(5b) $B \to b$
$C \to \cancel{abb} \mid SB$

(4)

$S \to AbA \longrightarrow \quad S \to Ab \mid$
$A \to aA \mid \epsilon$

$S \to AbA \mid bA \mid Ab \mid b$
$A \to aA \mid a$

$S \to CA \mid DA \mid AD \mid b$
$A \to BA \mid a$
$B \to a$
$C \to AD$
$D \to b$

Pumping Lemma

For every CFL L $\exists$ $p \in \mathbb{N}$ s.t. $\forall$ $w \in L$ with $|w| \geq p$
$\exists$ $x, y, z, u, v \in \Sigma^*$ s.t. $w = xyzuv$

(1) $|yu| \geq 1$
(2) $|yzu| \leq p$
(3) $xy^izu^iv \in L$

Not a CFL pumping lemma true

proof idea

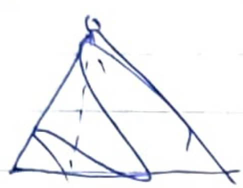
n length string

$\log_2 n \leq$ parse tree height

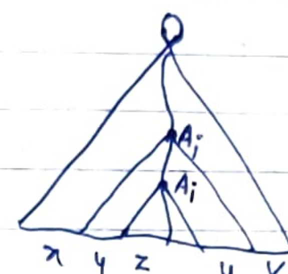
$\boxed{\text{Parse tree height} = k \quad \text{~~string~~ length} \leq 2^{k}-1}$

m is no of variables in CNF

$p = 2^m$



height = m+1



(3) $\{a^n b^n c^n \mid n \geq 0\}$

$w = a^p b^p c^p$

x =

y =

z =

u =

v =

yzu will lies in a's or b's or c's or ab or bc

pumping all will not increase

Pumping lemma does not hold

By statement of pumping lemma $|yzw| \leq p$

Therefore the strings ~~y and~~ yzw may contain only a's or only b's or only c's or atmost 2 symbols (a b) (b c)

$yu \neq \epsilon$

$\Rightarrow xy^2zu^2v$ increase atmost two symbols not all, $xy^2zu^2v \notin L$

(E) $L = \{ww \mid w \in (0+1)^*\}$

$0^p 1^p 0^p 1^p$

$xyzuv$

$yzu \in 0^p$

$yzu \in 1^p$

$yzu \in 1^p 0^p$

$xy^0 z u^0 v$

1100

Pumping:

reduce 0

reduce 1

reduce 0 & 1

(E) $L = \{a^i b^j c^k \mid i \leq j \leq k\}$

$a^p b^p c^{p+}$

(E) $\{a^p \mid p \text{ is a prime}\}$

$\{a^{n!} \mid n \geq 0\}$

$\{a^{2^n} \mid n \geq 0\}$