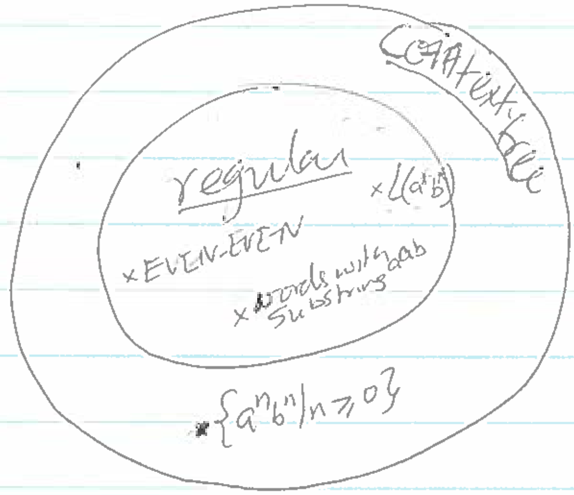
Ch. 3 Context-free languages



1. regular language
   - regular expressions
   - dfa
   - nfa

2. Context-free language
   - context-free gram (cfg)
   - pushdown automa (pda)

def: A context free grammar $G = (V, \Sigma, R, S)$
where V is an alphabet
$\Sigma \subseteq V$, is set of terminals
R : set of rules : finite subset of $(V-\Sigma) \times V^*$
$S \in V - \Sigma$, start symbol
$(A, u) \in R$, is written as $A \xrightarrow[G]{} u$
members of $V - \Sigma$ are called non-terminals

ex G: $V = \{S, A, B, M, a, b\}$
$V - \Sigma = \{S, A, B, M\}$, $\Sigma = \{a, b\}$

R :
$S \xrightarrow[G]{} aMb$
$M \xrightarrow[G]{} A$
$M \xrightarrow[G]{} B$
$A \xrightarrow[G]{} aA$
$A \xrightarrow[G]{} \epsilon$
$B \xrightarrow[G]{} bB$
$B \xrightarrow[G]{} \epsilon$

$a(a^* \cup b^*)b$

rewriting system
$S \Rightarrow aMb \Rightarrow aAb \Rightarrow aaAb$
$\Downarrow$
aab

G generates aab

for any $u, v \in V^*$, we write

$u \underset{G}{\Rightarrow} v$ iff there exists $x, y, v' \in V^*$, $A \in V - \Sigma$ such that $u = xAy$, $v = xv'y$ and $A \underset{G}{\rightarrow} v'$

$\overset{*}{\underset{G}{\Rightarrow}}$: reflexive transitive closure of $\underset{G}{\Rightarrow}$.

$$L(G) = \{ w \mid w \in \Sigma^* \text{ and } S \overset{*}{\underset{G}{\Rightarrow}} w \}$$

A language L is context free if it is equal to $L(G)$ for some CFG G.

$w_0 \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n$: derivation of w_n from w_0.

n = length of derivation.

Ex 3.1.4 G: $S \to AA$
$A \to AAA$
$A \to a$
$A \to bA$
$A \to Ab$

Claim: $L(G) = \{ w \mid w \in \{a,b\}^*$ and w has even ~~[illegible]~~ positive # a's $\}$ = EVENAs

Proof:

1) $L(G) \subseteq$ EVENAs.

We shall prove the following stronger fact:
if $w \in V^*$ and $S \stackrel{+}{\Rightarrow} w$ then (# A's + # a's) in w ~~#~~ is positive & even.

by induction on length, k of the derivation of w from S.

Basis Step: k=1
w can ~~be~~ only be AA since $S \Rightarrow AA$ is the only derivation of length = 1.
Clearly AA has (2+0) A's and a's which is positive & even.

Induction Step:
let $S \stackrel{*}{\Rightarrow} w$ in k or ~~less~~ fewer steps
then w has positive even # A's + a's — IH

Consider $S \stackrel{+}{\Rightarrow} w$ in k+1 steps
ie $S \stackrel{+}{\Rightarrow} w'$ in k steps
and $w' \Rightarrow w$ in one step.
w' has positive even # A's, a's.
w is derived from w' by application of one rule. each rule either adds 2 A's or ~~[illegible]~~ replaces A by a or does not change # A's or a's
∴ w has positive even # A's, a's.

2) EVENAs $\subseteq L(G)$.

Let $w \in$ EVENAs.

$$w = b^{m_1} a \, b^{m_2} a \cdots b^{m_{2n}} a \, b^{m_{2n+1}}$$

1^{st} a, 2^{nd} a, $(2n)^{th}$ a

$m_i \geq 0$

derivation for w:

$S \Rightarrow AA$ apply $S \to AA$ once

$\overset{*}{\Rightarrow} A^{2n}$ $A \to AAA$ $n-1$ times

$\overset{*}{\Rightarrow} b^{m_1} A^{2n}$ $A \to bA$ m_1 times

$\Rightarrow b^{m_1} a A^{2n-1}$ $A \to a$ once.

$\overset{*}{\Rightarrow} b^{m_1} a b^{m_2} A^{2n-1}$ $A \to bA$ m_2 times

$\Rightarrow b^{m_1} a b^{m_2} a A^{2n-2}$ $A \to a$ once

$\vdots$

$\Rightarrow b^{m_1} a b^{m_2} a \cdots b^{m_{2n}} a \, b^{m_{2n+1}}$

$\therefore$ $w \in L(G)$.

$\Sigma = \{ (,) \}$

Ex G: $S \to \epsilon$

$S \to SS$

$S \to (S)$

produces balanced parantheses

$()(()) \in L(G)$

$S \Rightarrow SS \Rightarrow S(S) \Rightarrow S((S)) \Rightarrow (S)(()) \Rightarrow ()(())$

another derivation

$S \Rightarrow SS \Rightarrow (S)S \Rightarrow ()S \Rightarrow ()(S) \Rightarrow ()((S)) \Rightarrow ()(())$

Note $L(G)$ is not regular because

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

not regular.

ex Palindromes	Even Palindromes	Odd Palindrom
$S \to aSa$	$S \to aSa$	$S \to aSa$
$S \to bSb$	$S \to bSb$	$S \to bSb$
$S \to \epsilon$	$S \to \epsilon$	$S \to a$
$S \to a$		$S \to b$
$S \to b$		

ex EQUAL is a C.F.L.
from Cohen

3.2 Regular Languages and CFLs

def A cfg $G = (V, \Sigma, R, S)$ is regular iff $R \subseteq (V - \Sigma) \times \Sigma^* ((V - \Sigma) \cup \{\epsilon\})$

i.e. all rules are of the form

$A \longrightarrow wB$ or ~~where~~

$A \longrightarrow w$

where A, B: nonterminals and $w \in \Sigma^*$.

---

ex G:
$$S \to bA$$
$$S \to aB$$
$$A \to abaS$$
$$B \to babS$$
$$S \to \epsilon$$

is regular.

$$L(G) = L((abab \cup baba)^*)$$!!

~~Clar~~ Th 3.2.1 A language is regular iff it can be ~~is~~ generated by a regular grammar:

Proof: (sketch)

only if part: Let L be regular

so there is a dfa $M = (K, \Sigma, \delta, s, F)$

s.t. $L = L(M)$

equivalent regular grammar:

$V = \Sigma \cup K$, $S = s$,

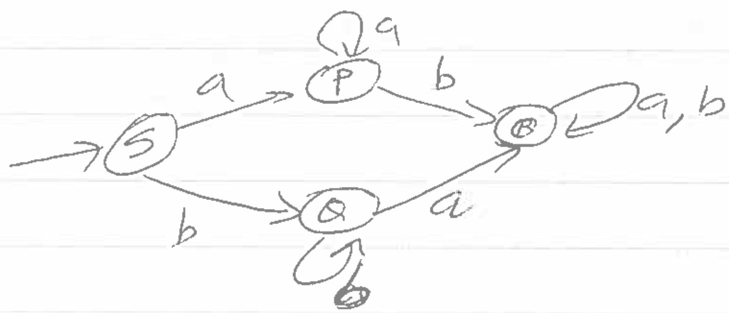
$R = \{ q \to ap \mid \delta(q,a) = p \} \cup$

$\{ q \to \epsilon \mid q \in F \}$

assume $\Sigma \cap K = \phi$.

ex

M:



CFG:

$S \to aP$

$S \to bQ$

$P \to aP$

$P \to bR$

$Q \to bQ$

$Q \to aR$

$R \to aR$

$R \to bR$

$R \to \epsilon$

$L(G) = L(M)$ ✓

<u>if part</u> Let $G = (V, \Sigma, R, S)$ be a ~~CF~~ regular grammar

Construct equivalent nfa as follows

M: $K = (V - \Sigma) \cup \{f\}$ $\quad f$: new state

$s = S$

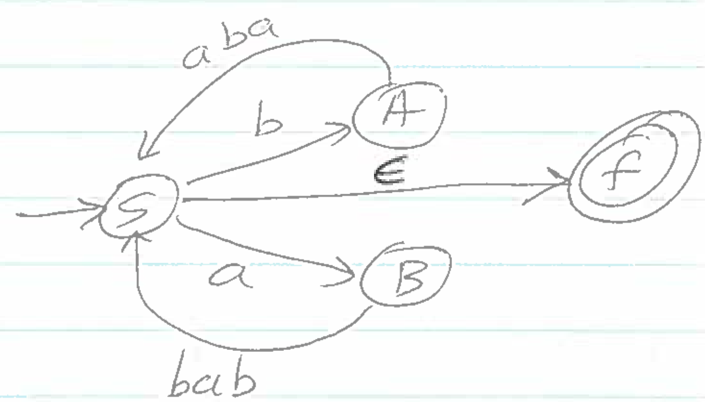
$F = \{f\}$

$\delta = \{(A, w, B) \mid A \to wB \in R\} \cup$

$\{(A, w, f) \mid A \to w \in R\}$

$L(M) = L(G)$ ✓

<u>ex</u> G: $S \to bA$
$S \to aB$
$A \to abaS$
$B \to babS$
$S \to \epsilon$



3.3 Pushdown Automata (PDA)

Def: A PDA M consists of six parts $(K, \Sigma, \Gamma, \Delta, s, F)$ where

K = finite set of states

Σ = alphabet of input symbols

Γ = alphabet of stack symbols

$s \in K$ = initial state

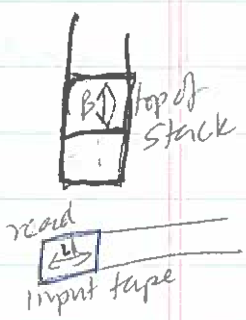
$F \subseteq K$ = set of final states

Δ : transition relation finite subset of $(K \times \Sigma^* \times \Gamma^*) \times (K \times \Gamma^*)$

non-determinism

$((p, u, \beta), (q, \gamma)) \in \Delta$ means that M

in state p ~~reads~~ with β on top of stack reads input u, replaces β by γ on stack and goes to state q.



Note: $\boxed{\text{nondeterministic machine}}$

$((p, u, \epsilon), (q, a))$ pushes 'a' on stack ; read u

$((p, u, a), (q, \epsilon))$ pops 'a' from stack ; read u.

Configuration : element of $K \times \Sigma^* \times \Gamma^*$

(q, w, α) : in state q.
remaining input : w
stack (read top-down) : α

ex (q, w, abc)
top

define: $(p, ux, \beta\alpha) \vdash_M (q, x, \gamma\alpha)$ — yields in one step

iff $((p,u,\beta),(q,\gamma)) \in \Delta$

define $\vdash_M^*$ as the ref. tr. closure of $\vdash_M$.

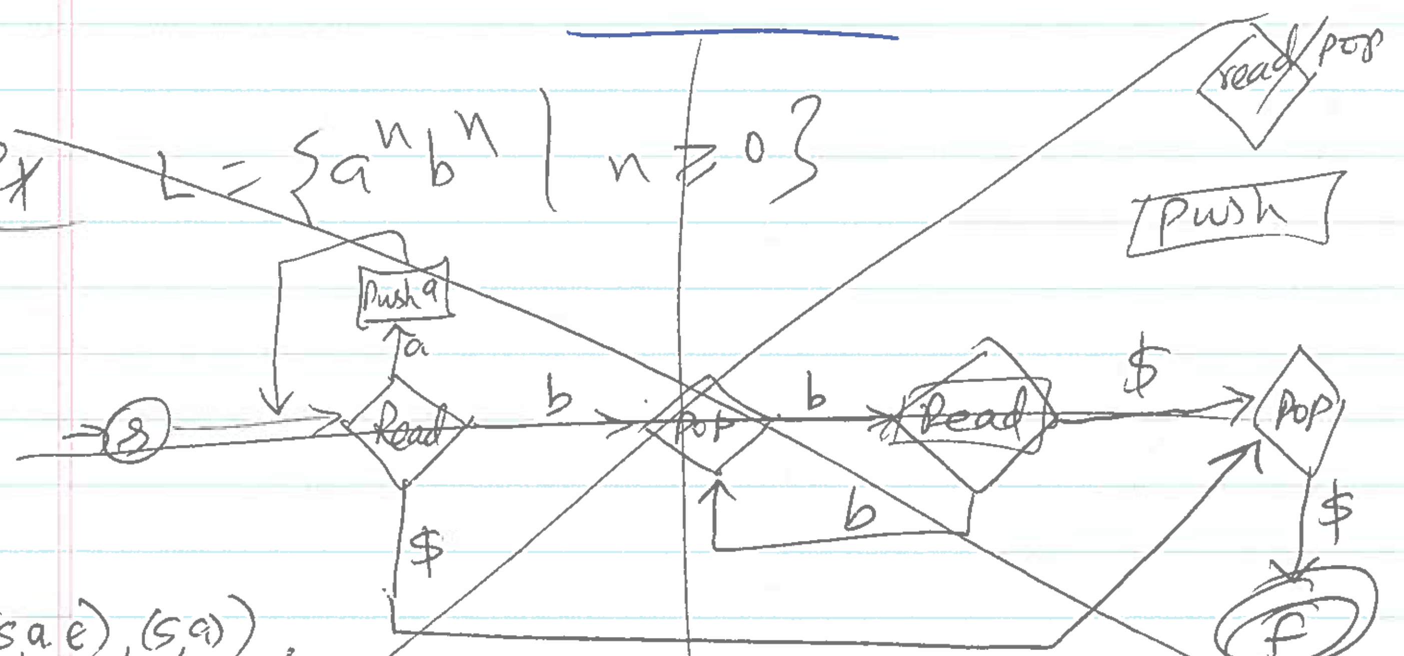
def M **accepts** $w \in \Sigma^*$ **iff**

$(s, w, \epsilon) \vdash_M^* (p, \epsilon, \epsilon)$ for some $p \in F$.

empty stack — empty stack

def $L(M) = \{ w \mid w \in \Sigma^* \text{ and } M \text{ accepts } w \}$

ex $L = \{a^n b^n \mid n \geq 0\}$



$\Delta = \{((s,a,\epsilon),(s,a)),$
$((s,b,a),(f,\epsilon))$
$((f,b,a),(f,\epsilon))\}$ $F = \{q\}$ $\Gamma = \{a,b\}$ $\Sigma = \{a,b\}$, $K = \{s,f\}$

Note Every f.a. can be viewed as a PDA that never operates on its stack. Let $M = (K, \Sigma, \Delta, s, F)$ be a f.a. Construct $M' = (K, \Sigma, \Gamma, \Delta', s, F)$ PDA

$$\Gamma = \phi$$
$$\Delta' = \{ ((p,u,\epsilon),(q,\epsilon)) \mid (p,u,q) \in \Delta \}$$

Ex $L = \{ wcw^R \mid w \in \{a,b\}^* \}$

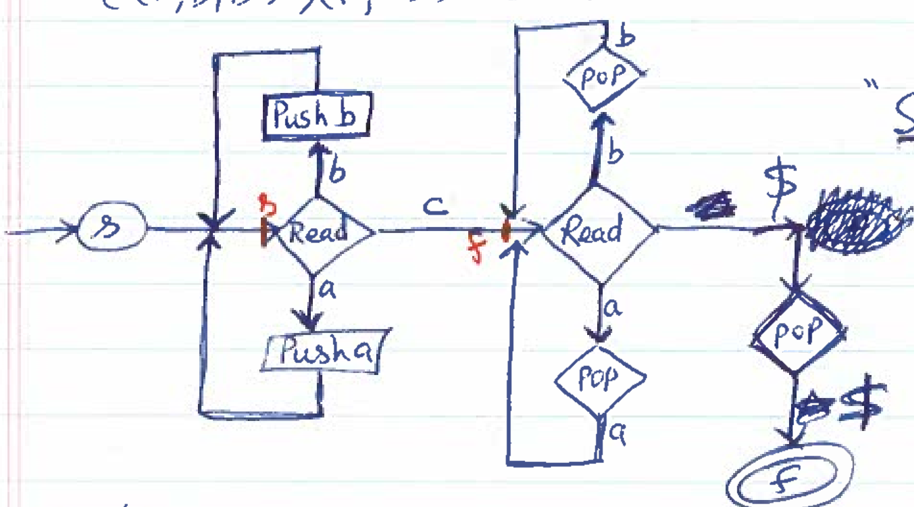
$M = (K, \Sigma, \Gamma, \Delta, s, F)$; $K = \{s, f\}$

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

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

$\Delta =$ $\qquad F = \{f\}$

$\{ ((s,a,\epsilon),(s,a)),$ ← read and push a; b
$((s,b,\epsilon),(s,b)),$ ←
$((s,c,\epsilon),(f,\epsilon)),$ ← read c and go to f.
$((f,a,a),(f,\epsilon))$ ← read, pop same letter.
$((f,b,b),(f,\epsilon)) \}$ ←

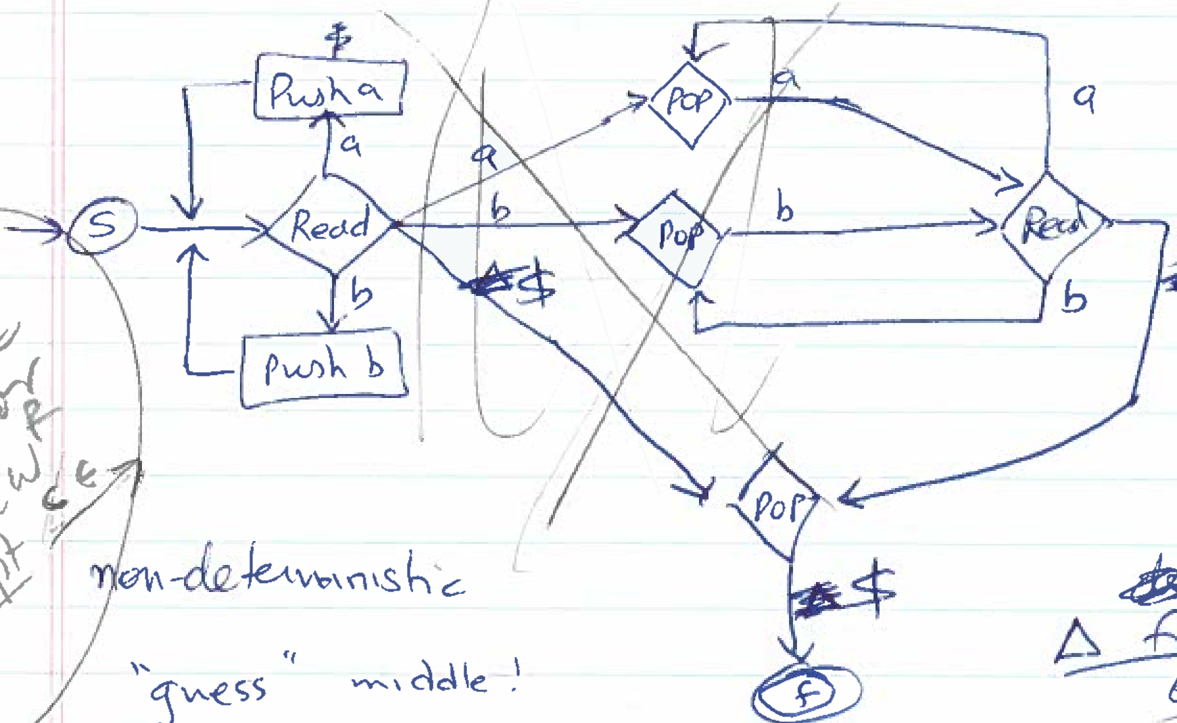


"State diagram"

$: symbol after last position in input or symbol below bottom of stack

try abbcbba

Ex $L = \{ww^R \mid w \in \{a,b\}^*\}$ EVEN PALINDROM



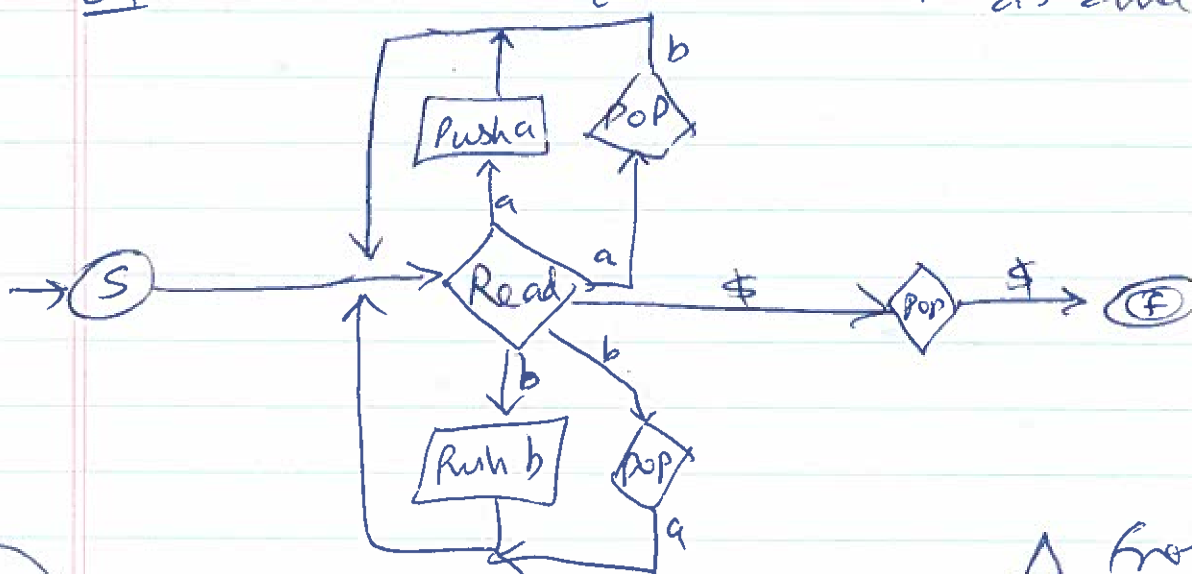
Same as for wcw^R except c

non-deterministic

"guess" middle!

Δ from p109 Ex 3.3.3

Ex EQUAL $= \{w \in \{a,b\}^* \mid w$ has equal # a's and b's $\}$



Δ from p109 Ex 3.3.4

p493 Cohen

111

3.4 PDA and Cfgs

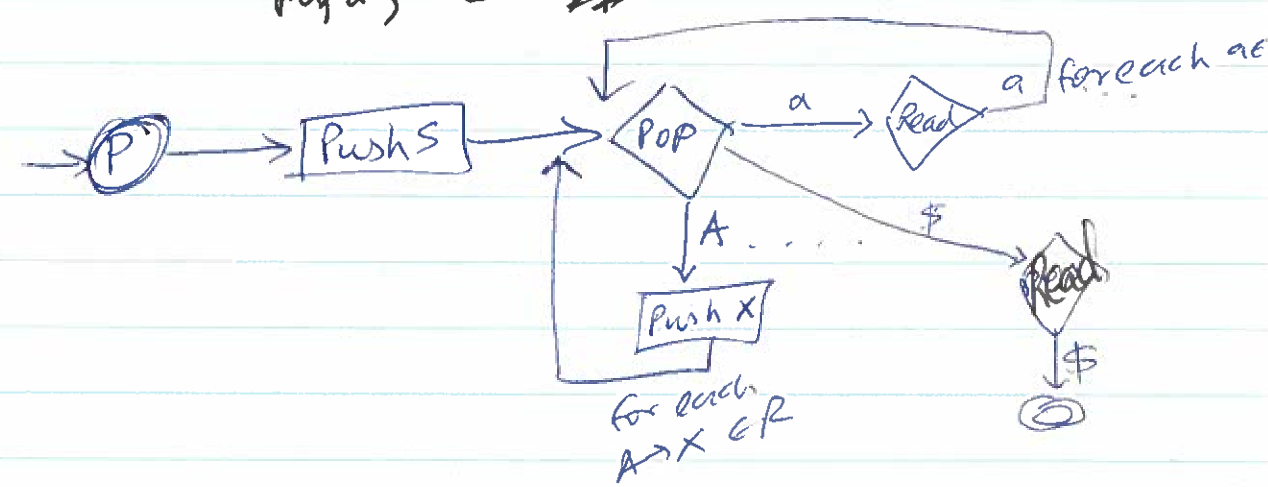
Lemma 3.4.3 Every CFL is accepted by some PDA.

Proof let $G = (V, \Sigma, R, S)$ be a CFG;
we need to construct PDA, M s.t. $L(M) = L(G)$

Construct $M = (\{p, q\}, \Sigma, V, \Delta, p, \{q\})$
where Δ contains the following transitions:

1) $((p, \epsilon, \epsilon), (q, S))$ Push S
2) $((q, \epsilon, A), (q, x))$ Pop A, push x for each $A \to x$ in R
3) $((q, a, a), (q, \epsilon))$ for each $a \in \Sigma$.
pop 'a', read 'a'

i.e



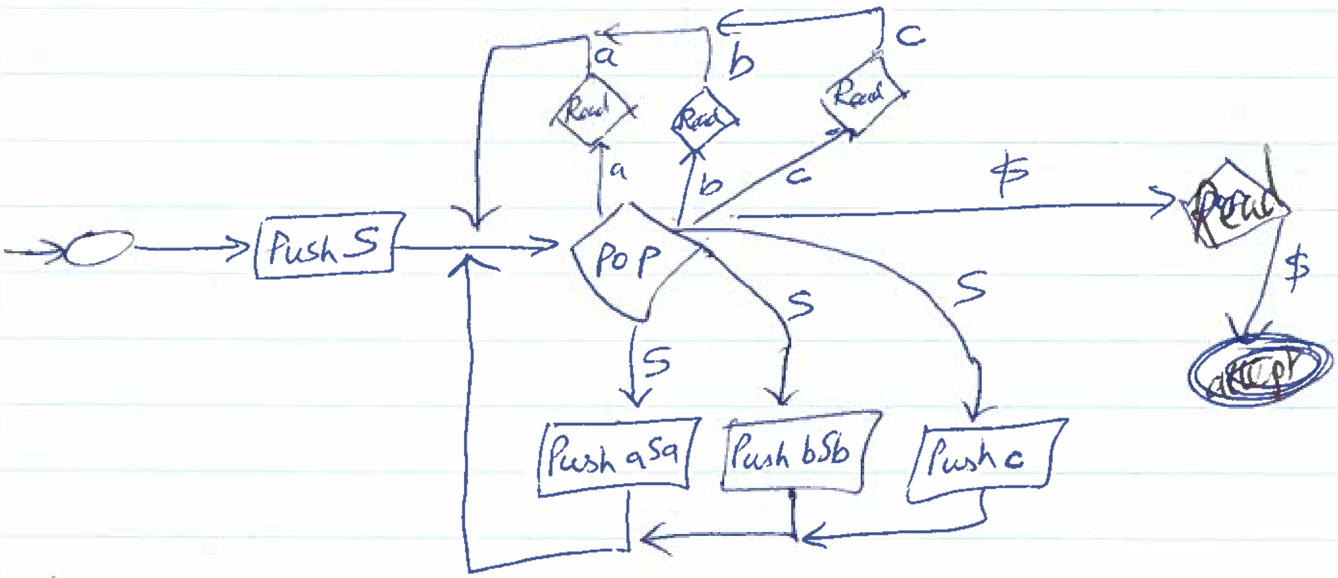
$\Sigma = \{a, b, c\}$

ex CFG

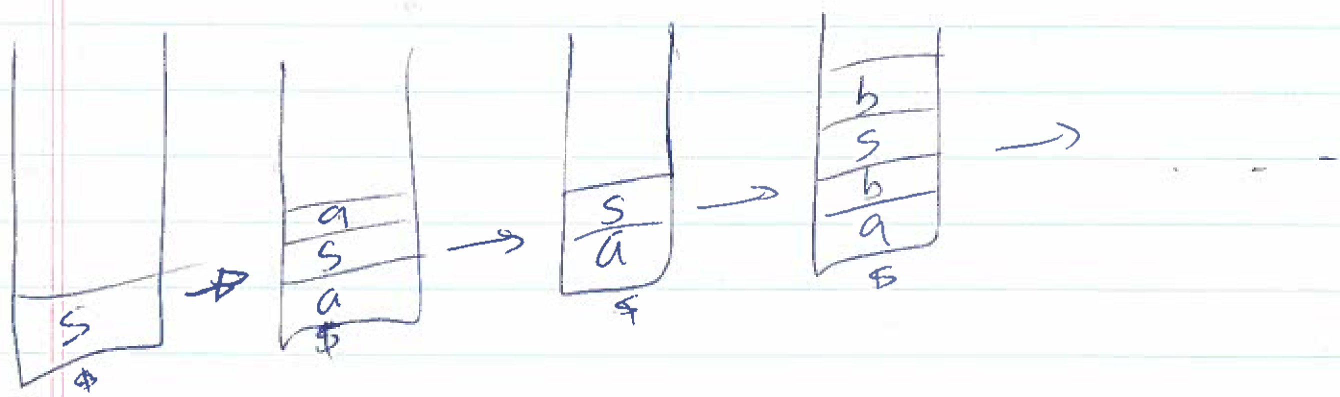
$S \rightarrow aSa$

$S \rightarrow bSb$

$S \rightarrow c$



abbcbba$



Lemma 3.4.4 If a language is accepted by a PDA then it is a context-free language

Proof Skip it.

3.5 Properties of CFLs

1. Closure Properties

Th. 3.5.1 CFls are closed under union, concatenation, and Kleene star

Proof:

Let $G_1 = (V_1, \Sigma_1, R_1, S_1)$ and

$G_2 = (V_2, \Sigma_2, R_2, S_2)$

wlg, assume $V_1 - \Sigma_1 \cap V_2 - \Sigma_2 = \phi$

I) Union: $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1, S \to S_2\}, S)$

new symbol

$L(G) = L(G_1) \cup L(G_2)$.

II) Concatenation: $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1 S_2\}, S)$

new symbol

$L(G) = L(G_1) L(G_2)$

III) Kleene Star:

$G = (V_1, \Sigma_1, R_1 \cup \{S_1 \to \epsilon, S_1 \to S_1 S_1\}, S_1)$

$L(G) = L(G_1)^*$

Do Some examples

CFls not closed under $\cap$, complement.
(we shall see later).

Th. 3.5.2: The intersection of a CFL and a regular language is a context-free language

Proof: (based on f.a & pda)

- L: CFL ; $M_1 = (K_1, \Sigma_1, \Gamma_1, \Delta_1, s_1, F_1)$: PDA
  s.t. $L(M_1) = L$
- R: reg. lang. ; $M_2 = (K_2, \Sigma_2, \delta_2, s_2, F_2)$: dfa
  s.t. $L(M_2) = R$

Combine these machines into PDA to simulate action of both.

PDA $M = (K, \Sigma, \Gamma, \Delta, s, F)$

$K = K_1 \times K_2$

$\Sigma = \Sigma_1 \cup \Sigma_2$

$\Gamma = \Gamma_1$

$s = (s_1, s_2)$

$F = F_1 \times F_2$

Δ: $\left(((q_1, q_2), u, \beta), ((p_1, p_2), \gamma) \right) \in \Delta$

iff $((q_1, u, \beta), (p_1, \gamma)) \in \Delta_1$ and

$(q_2, u) \vdash^*_{M_2} (p_2, \epsilon)$

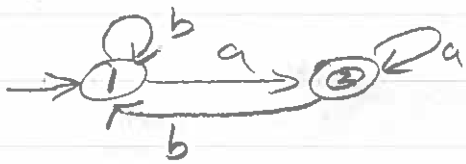
Th. 3.5.2 example

CFL: $L_1 = \text{EQUAL}$

$\Sigma = \{a, b\}$
$\Gamma = \{a, b\}$
$F_1 = \{s\}$

$\Delta = \{ ((s, a, \epsilon), (s, a)),$
$((s, a, b), (s, \epsilon)),$
$((s, b, \epsilon), (s, b)),$
$((s, b, a), (s, \epsilon)) \}$

regular: L_2 = words ending in letter 'a'.



$\delta(1, a) = 2$
$\delta(1, b) = 1$
$\delta(2, a) = 2$
$\delta(2, b) = 1$

$F_2 = \{2\}$

$L_1 \cap L_2$: $K = K_1 \times K_2 = \{(s, 1), (s, 2)\}$

$\Sigma = \{a, b\}$
$\Gamma = \{a, b\}$
$s = (s, 1)$ call it $s1$
$F = F_1 \times F_2 = \{(s, 2)\} = \{s2\}$

$\Delta = \{ ((s1, a, \epsilon), (s2, a)),$
$((s1, a, b), (s2, \epsilon)),$
$((s1, b, \epsilon), (s1, b)),$
$((s1, b, a), (s1, \epsilon)),$

$((s2, a, \epsilon), (s2, a)),$
$((s2, a, b), (s2, b)),$
$((s2, b, \epsilon), (s1, b))$
$((s2, b, a), (s1, \epsilon)) \}$

ex $abba \in L_1 \cap L_2$

$$
\begin{aligned}
(q1, abba, \epsilon) &\vdash (q2, bba, a) \\
&\vdash (q1, ba, \epsilon) \\
&\vdash (q1, a, b) \\
&\vdash (\underset{\substack{\uparrow \\ \text{final}}}{q2}, \epsilon, \epsilon)
\end{aligned}
$$

~~Example:~~

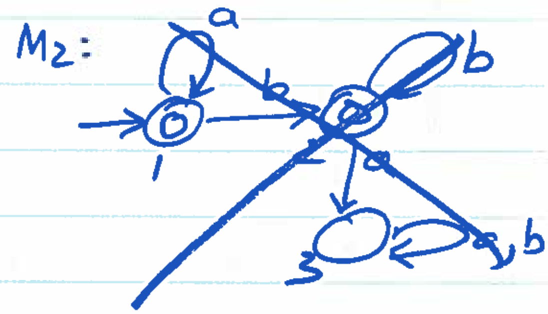
~~regular~~

~~$\text{EQUAL} \cap L(a^*b^*) = \{a^n b^n \mid n \ge 1\}$~~

~~M_1:~~

~~$\Delta_1 : \{ ((q,\epsilon,\epsilon),(q,c)), ((q,a,c),(q,ac)), ((q,a,a),(q,aa)), ((q,a,b),(q,\epsilon))$~~

~~M_2:~~



Ex $L = \{ w \mid w \in \{a,b\}^*$ **and** w has equal # a's & b's **and** w does not contain substrings $abaa$ and $babb \}$

$L = \text{EQUAL} \cap L_1$

where $L_1 = L((a \cup b)^*) - L((a\cup b)^*(abaa \cup babb)(a \cup b)^*)$

L_1 is regular

EQUAL is c.f.

$\therefore L$ is c.f.

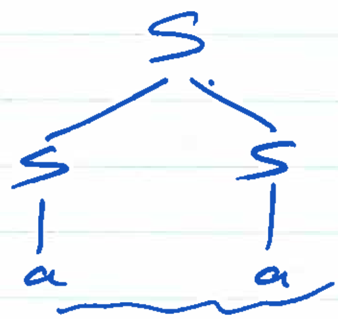
3.5.20

Periodicity Properties

Parse trees: Ex: G: $S \to SS$
$S \to a$

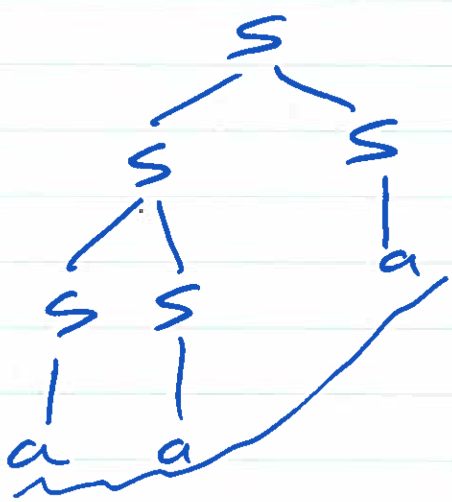
$S \Rightarrow SS \Rightarrow aS \Rightarrow aa$

$S \Rightarrow SS \Rightarrow Sa \Rightarrow aa$



yield = aa

$S \Rightarrow SS \Rightarrow Sa \Rightarrow SSa \Rightarrow aSa \Rightarrow aaa$



yield = aaa

path in a parse tree is a sequence of distinct nodes ~~each connected to~~ from root to a leaf.

length = # edges in the path

height = length of longest path

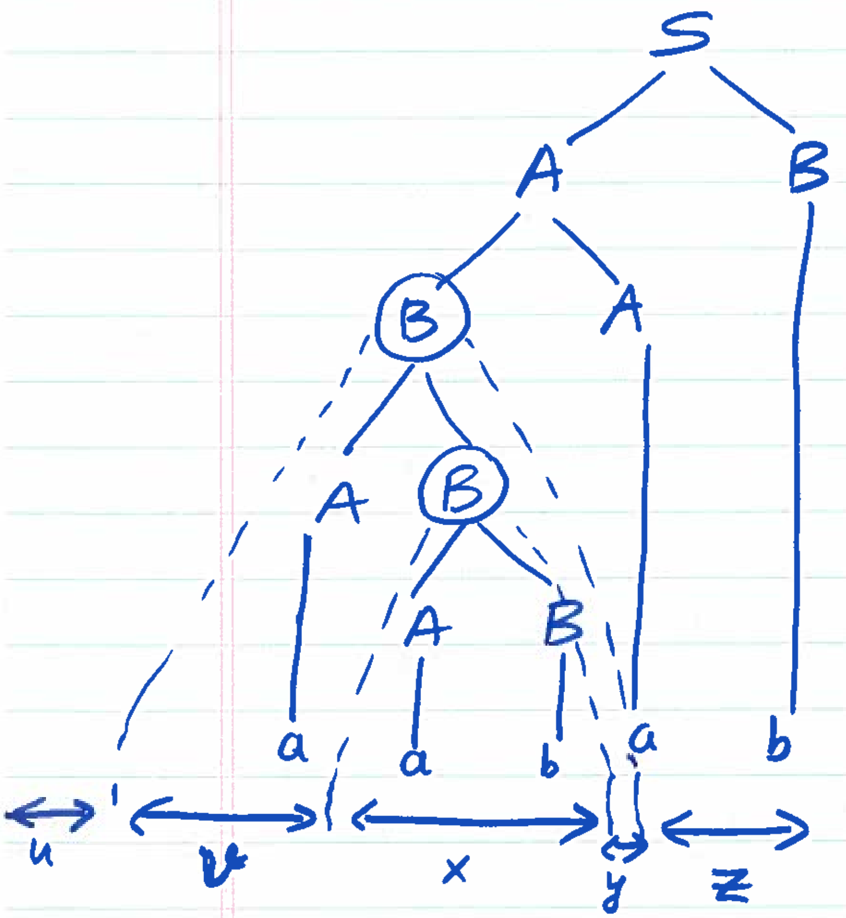
<u>Ex</u>

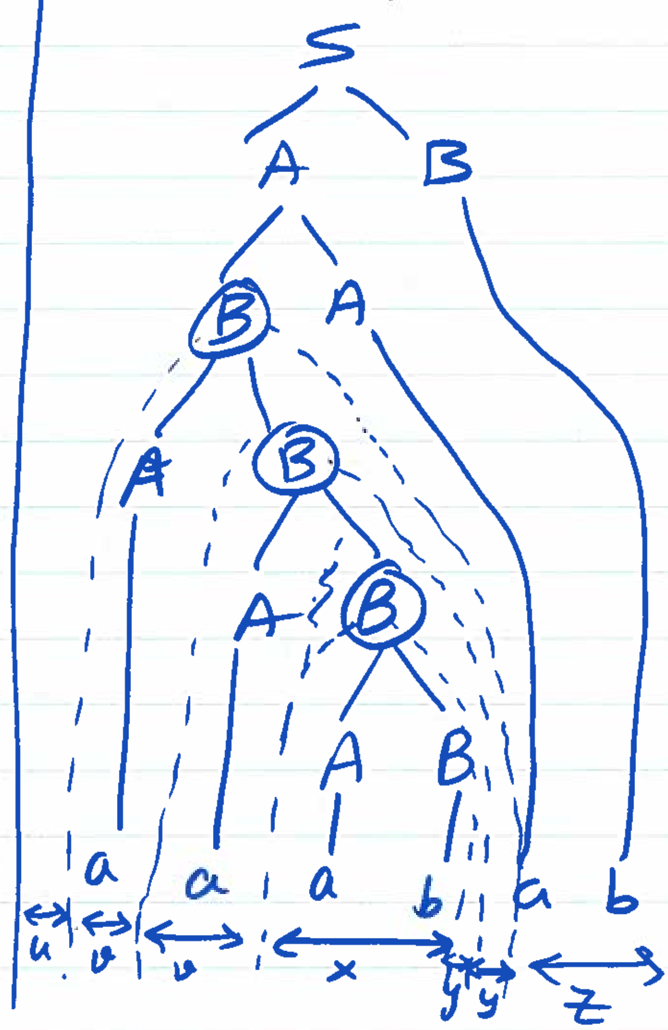
$S \to AB$
$S \to BC$
$A \to BA$
$C \to BB$
$B \to AB$

$A \to a$
$B \to b$
$C \to b$

SKIP

$w = aabab$

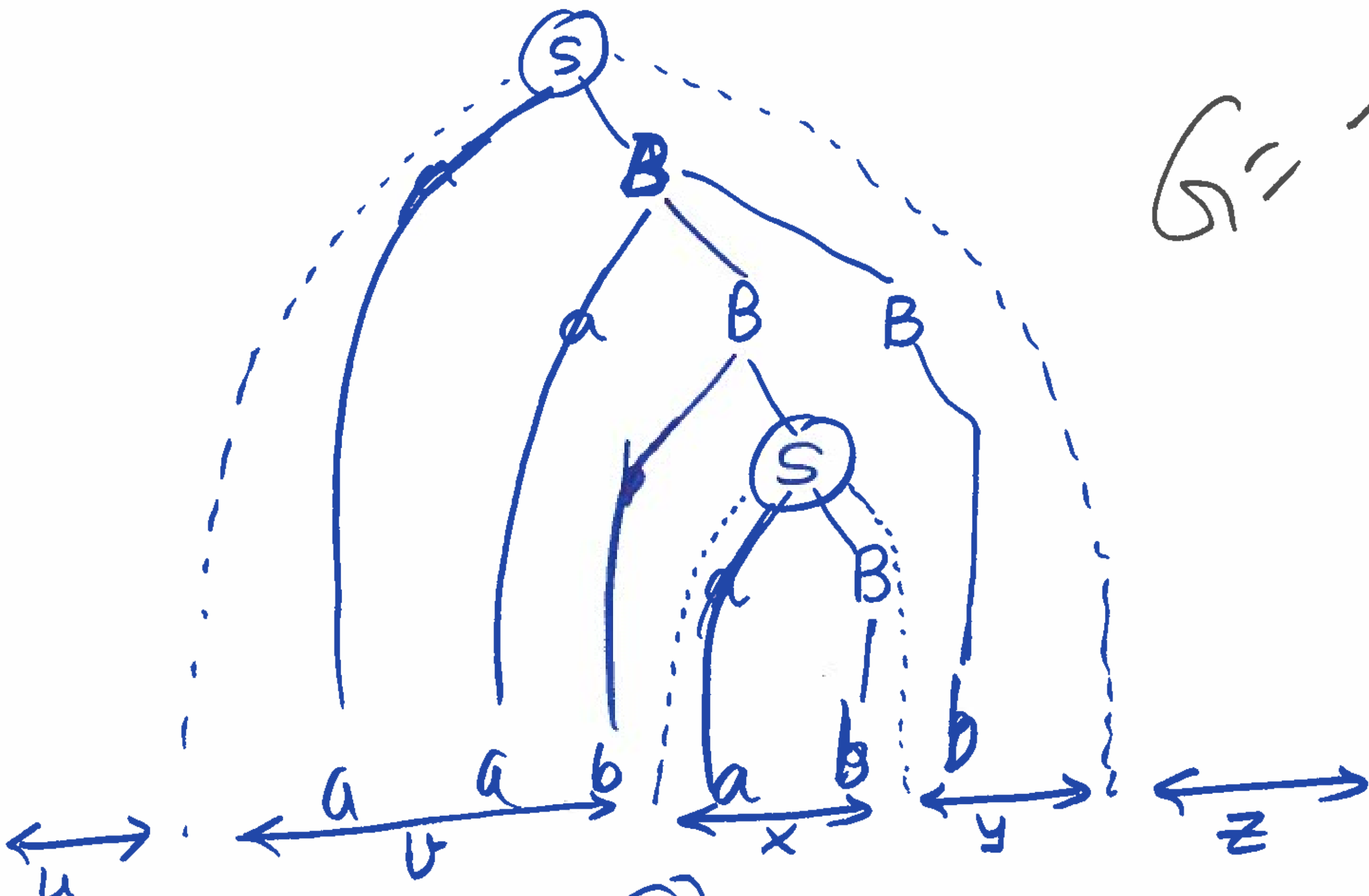


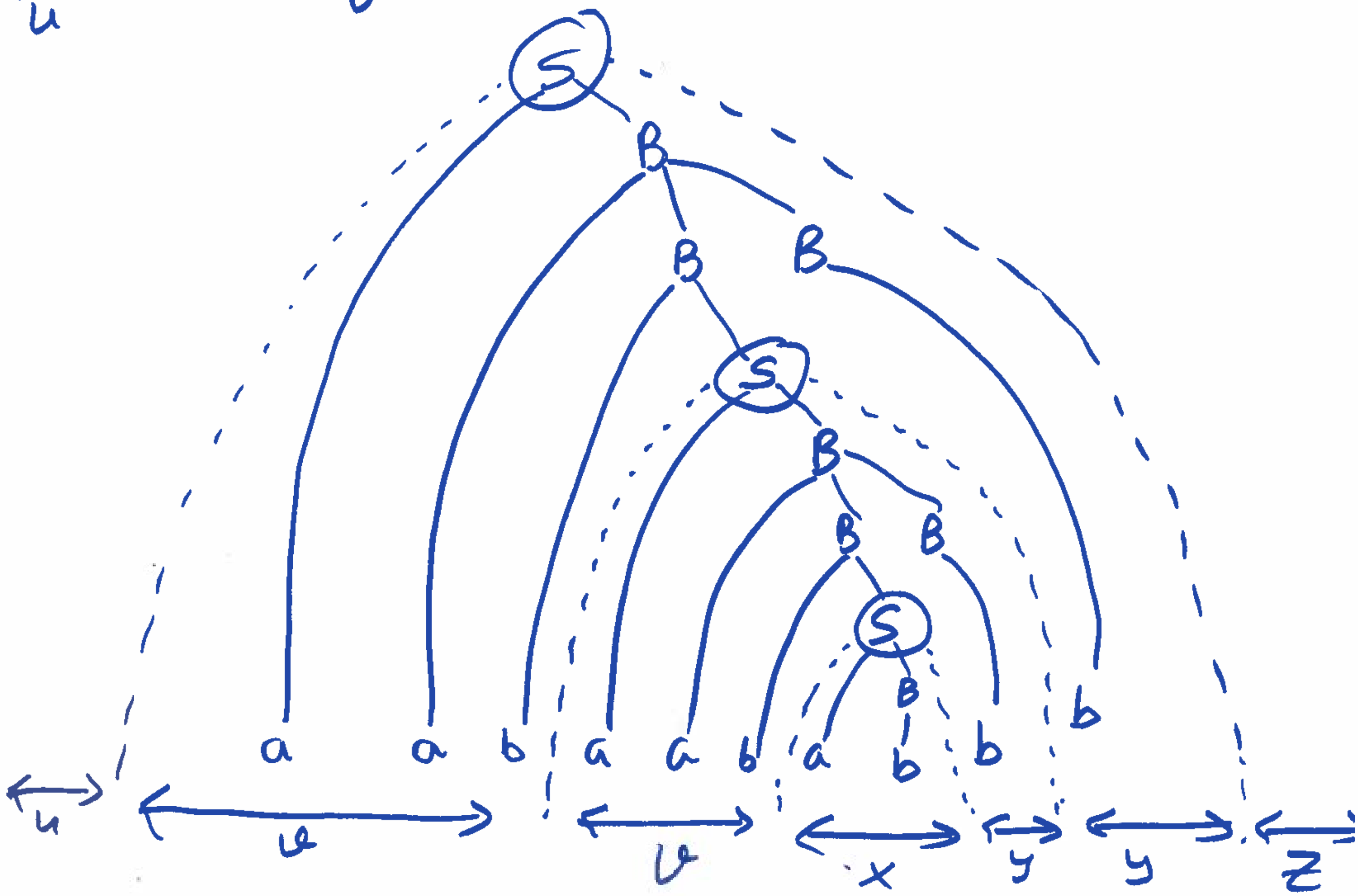


aababb

EQUAL

G = '

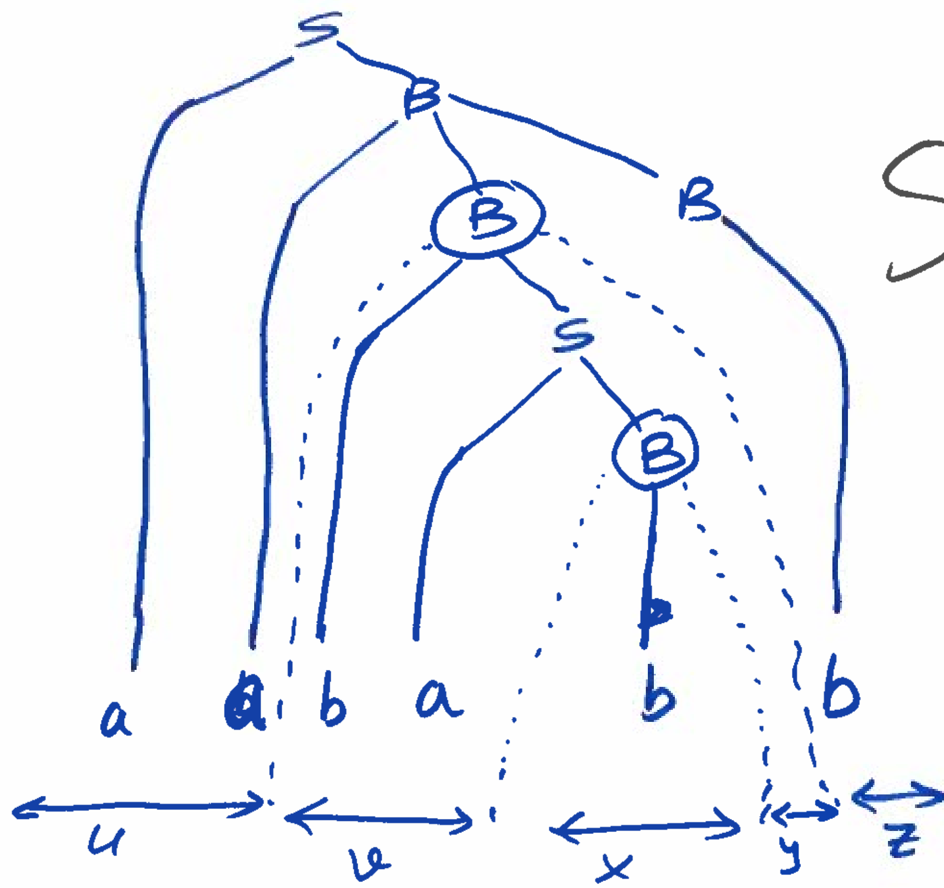




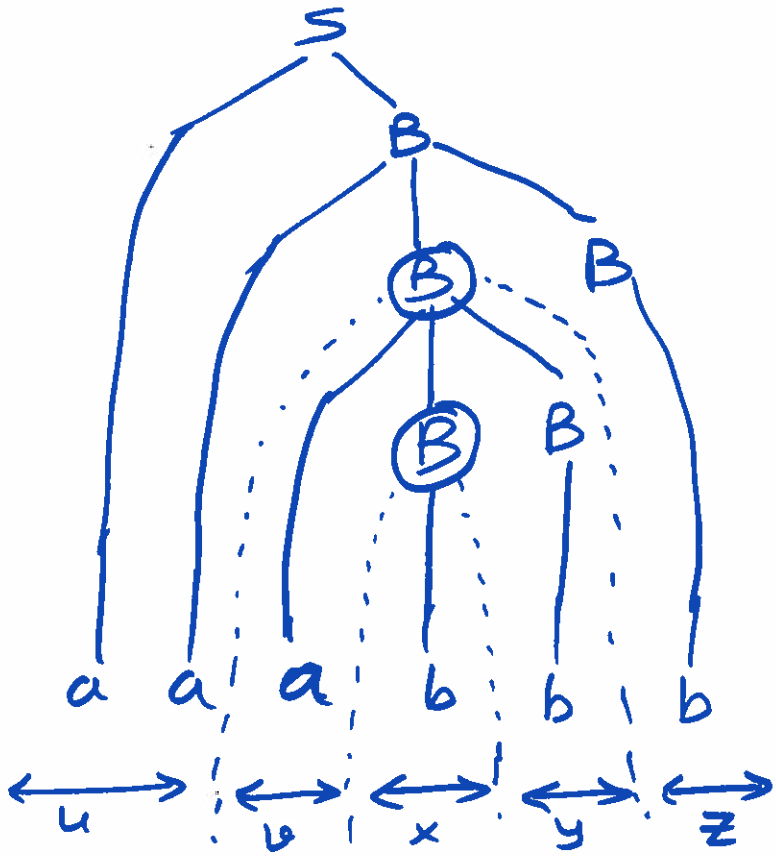
EQUAL

24

SKIP



EQUAL 25



$S \to aB$
$S \to bA$

$A \to aS$
$A \to bAA$
$A \to a$

$B \to bS$
$B \to aBB$
$B \to b$

SKIP

ex

$S \to PQ \qquad P \to a$

$Q \to QS \qquad Q \to b$

$w = abab$

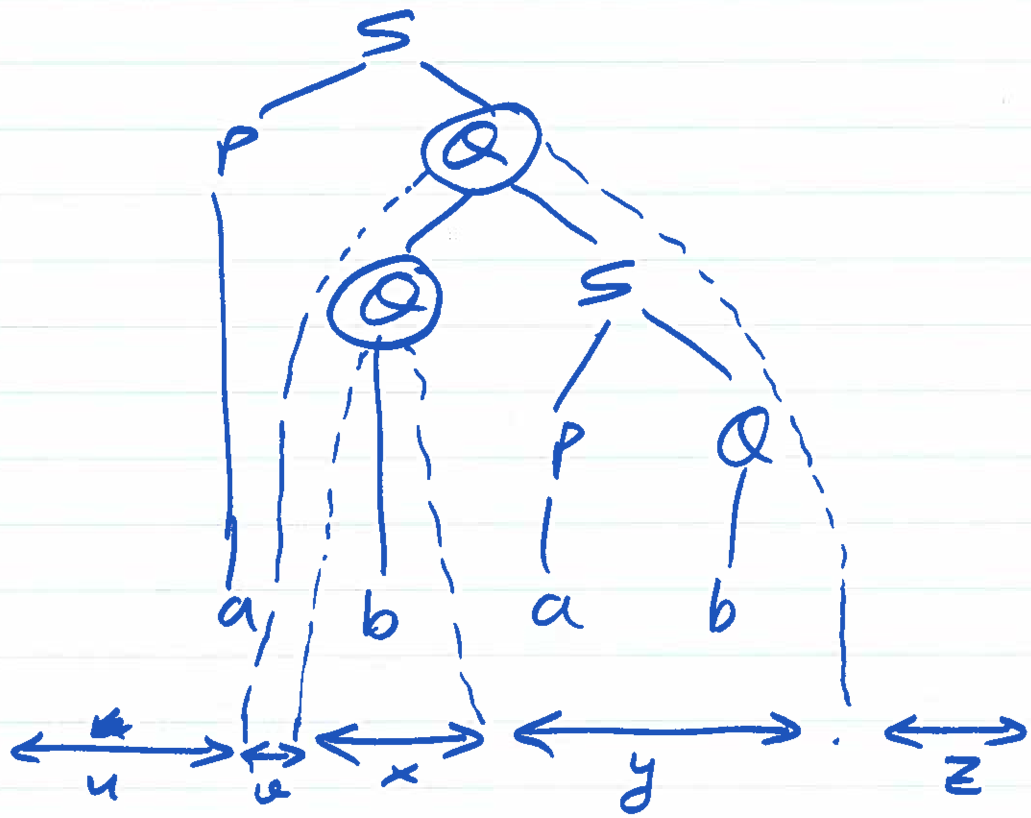
$u = a$

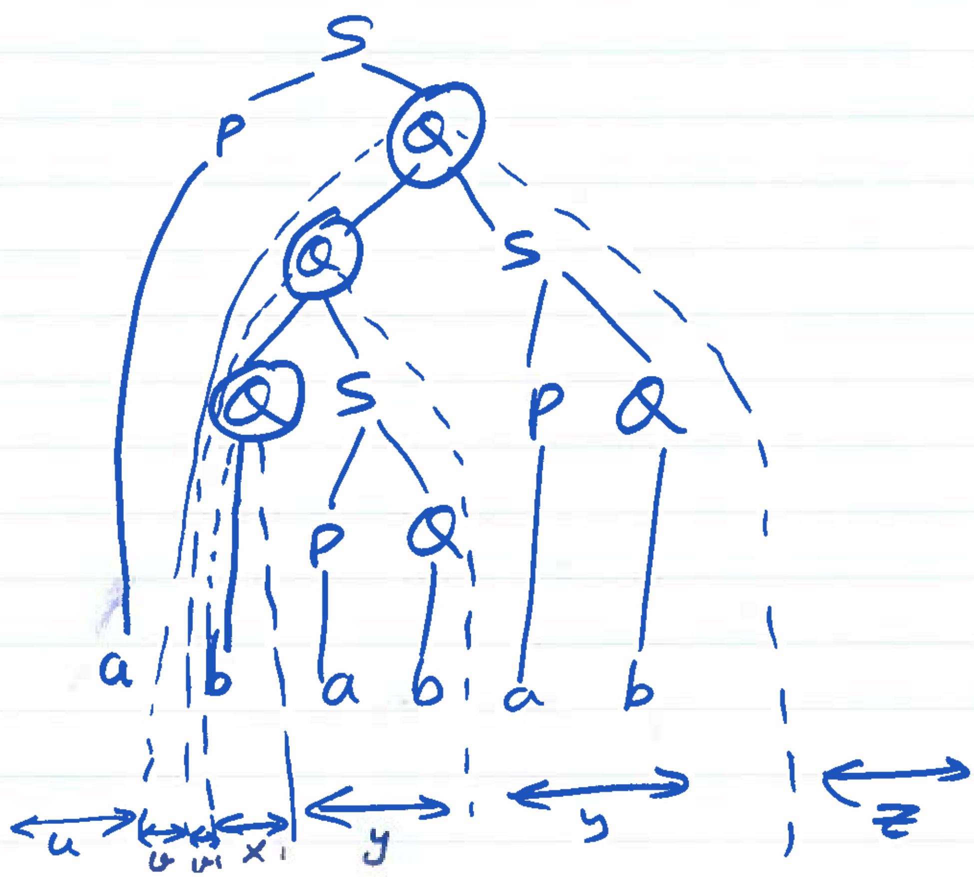
$v = \epsilon$

$x = b$

$y = ab$

$z = \epsilon$





Th. 3.5.3 (The Pumping Lemma).
Let G be a CFG. Then, there is a number k (depending on G) such that

if $w \in L(G)$ and $|w| > k$ then

$w = uvxyz$ for some u, v, x, y, z

where

1) $|vy| \geq 1$

and 2) $uv^i \cancel{x} y^i z \in L(G)$, $i \geq 0$

SKIP PROOF

Proof Let $G = (V, \Sigma, R, S)$

We need to show that there exists k s.t.
$w \in L(G)$ and $|w| > k$ implies

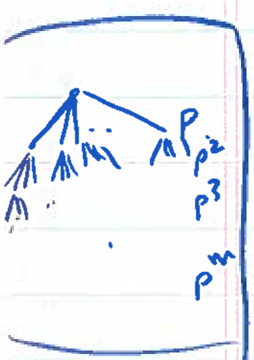
$$S \overset{*}{\Rightarrow} uAz \overset{*}{\Rightarrow} uvAyz \overset{*}{\Rightarrow} uvxyz$$

where $u, v, x, y, z \in \Sigma^*$, $A \in V - \Sigma$ $|vy| \neq 0$.

(so $A \overset{*}{\Rightarrow} vAy$ can be repeated many times.

Let p be the longest # symbols on the r.h.s of any rule in R i.e

$$p = \text{MAX}\{ |\alpha| \mid A \to \alpha \in R \}$$



A parse tree of height m can have at most p^m leaves. i.e If T is a parse tree with yield (# leaves) greater than p^m, then T has a path of length $> m$.

Let $m = |V - \Sigma|$ (# non-terminals)

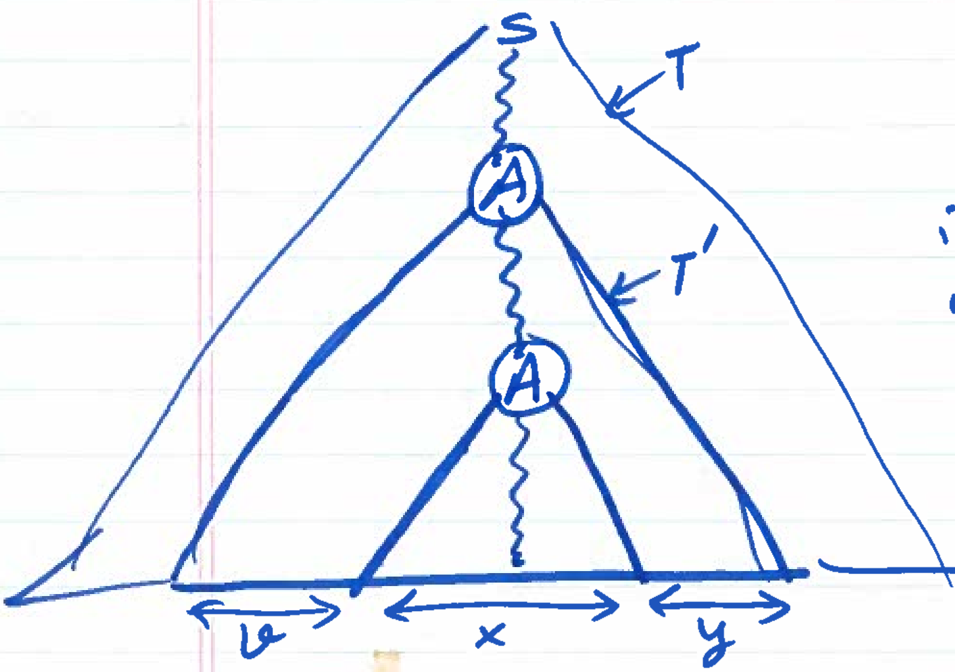
and $k = p^m$.

Suppose $|w| > k$.

T:

Since $|w| > k$; T has at least one path of length $> m$ (i.e # nodes in path is $> |V-\Sigma|+1$)

$\therefore$ There are two nodes on this path labeled by same nonterminal,



$v = y = \epsilon$ is not possible because if it were so, then we could replace T' by subtree starting with 2^{nd} A without changing the yield. This way we will end up in a parse tree with same yield but ~~[illegible]~~ height

NOT POSSIBLE

Stronger version of P.L.

Let L be a CFL. Then there exists K such that

if $w \in L$ and $|w| > K$ then

$w = uvxyz$ where

1) $|x| \geq 1$,
2) $|vy| \geq 1$,
3) $|vxy| \leq K$, and
4) $uv^ixy^iz \in L$ $i \geq 0$.

Th. 3.5.4 $L = \{a^n b^n c^n \mid n \geq 0\}$ is not a CFL

Proof: Suppose L is a CFL.

Then by P.L there is a K.

Consider, $w = a^K b^K c^K \in L$; clearly $|w| > K$.

$\therefore w = uvxyz$ where 1) $|x| \geq 1$. 2) $|vy| \geq 1$. 3) $|vxy| \leq K$ and 4) $uv^ixy^iz \in L$ $i \geq$

Possibilities for v, y:

1) Either v or y has 'ab' or 'bc' as substring

~~On pumping we get a word~~ uv^2xy^2w is a word in L in which 'ba' or 'cb' is a substring. Contradiction.

2) Or v and y ~~same letter~~ are words made up of ~~a single~~ only a's or only b's or only c's.

So $uv^2xy^2z \in L$ ~~but~~ has unequal # a's b's c's

Contradiction.

Th. 3.5.5 CFLs are not closed under intersection and complementation.

Proof

Intersection:

$L_1 = \{a^n b^n c^m \mid m, n \geq 0\}$

is a CFL $\because$

$S \to AB$

$A \to aSb$

$A \to \epsilon$

$B \to cB$

$B \to \epsilon$

$L_2 = \{a^m b^n c^n \mid n, m \geq 0\}$ is a CFL

$\because$ $S \to AB$

$A \to aA$

$A \to \epsilon$

$B \to bBc$

$B \to \epsilon$

$L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 0\}$

which is not a CFL.

Complementation: If CFL are closed under complementation, then since

$L_1 \cap L_2 = \Sigma^* - ((\Sigma^* - L_1) \cup (\Sigma^* - L_2))$

CFLs would also be closed under $\cap$.!!

Claim: let $L = \{a^n b^n a^n \mid n \geq 0\}$.
Then, $\bar{L}$ is c.f.

SKIP

Proof (P 486 cohen)

Define

$M_{pq} = \{a^p b^q a^r \mid p > q,\ p \geq 1, q \geq 1, r \geq 1\}$

$$\boxed{\begin{array}{l} S \to AXA \\ X \to aXb \\ X \to ab \\ A \to aA \\ A \to a \end{array}}$$

cf $M_{qp} = \{a^p b^q a^r \mid q > p,\ p \geq 1, q \geq 1, r \geq 1\}$

cf $M_{pr} = \{a^p b^q a^r \mid p > r,\ p \geq 1, q \geq 1, r \geq 1\}$

cf $M_{rp} = \{a^p b^q a^r \mid r > p,\ p \geq 1, q \geq 1, r \geq 1\}$

cf $M_{qr} = \{a^p b^q a^r \mid q > r,\ p \geq 1, q \geq 1, r \geq 1\}$

cf $M_{rq} = \{a^p b^q a^r \mid r > q,\ p \geq 1, q \geq 1, r \geq 1\}$

cf $M = \Sigma^* - L(aa^*bb^*aa^*)$

cf

Let $L_1 = \underbrace{M_{pq} \cup M_{qp} \cup M_{pr} \cup M_{rp} \cup M_{qr} \cup M_{rq} \cup M}_{\text{c.f.}}$

$\bar{L_1} = L$!! why?

not c.f

(1) all words which are not of the form $a^p b^q a^r$ are in M, so they are not in L_1.
i.e all words in $\bar{L_1}$ are of the form $a^p b^q a^r$

(2) if $p > q$ then such a word is in M_{pq}
if $q > p$ ——— M_{qp}
⋮
So, only possibility is $p = q = r$.

3.5.3 Algorithmic Properties

Th. 3.5.8 There are algorithms to answer the following questions about CFGs

(a) $w \in L(G)$?
(b) $L(G) = \phi$?

Proof:

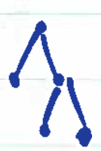
(a) ~~Before we answer $w \in L(G)$? question consider we need some definitions~~!
Suppose every rule in G is of the form

$$A \to u$$

where either (1) u is a terminal
or (2) $|u| \geq 2$.

Then, ~~every~~ the derived string will ~~never decrease in~~ become longer and longer except when $A \to a$ is used.

In such a situation, any parse tree of height h must have yield of length $\geq h$



So, if we check for all parse trees of height $\leq |w|$, we ~~would~~ can tell if there is a parse tree for w or not i.e if $w \in L(G)$ or not.

#such trees is finite!

We have to eliminate all rules of the form $A \to \epsilon$ and
$A \to B$

(we may have to keep $S \to \epsilon$!)

1) Eliminate rules of the form $A \to \epsilon$

Step 1. if $A \to \epsilon \in R$ and $B \to uAv \in R$
then introduce $B \to uv$ in R.

Step 2: Remove $A \to \epsilon$ (except when $A = S$!)

ex

G:
$S \to aAB$
$A \to \epsilon$
$B \to bAAb$
$A \to a$

$\Rightarrow$

G':
$S \to aAB$
$S \to aB$
$B \to bAb$
$B \to bb$
$B \to bAAb$
$A \to a$

$L(G) = L(G')$.

2) Eliminate rules of the form $A \to B$

1. Eliminate: $A \to \epsilon$

def: A nonterminal N is **nullable** if
1) $N \to \epsilon \in R$ **or**
2) $N \overset{*}{\Rightarrow} \epsilon$

Step1: **Delete** all rules of the form $A \to \epsilon$
step2: **Add** the following rules:
for every $X \to \alpha_1 N \alpha_2$ where N is nullable, add
$$X \to \alpha_1 \alpha_2$$
(exception: do not add $X \to \epsilon$)
if more than one occurrence of N on r.h.s **then** replace N by ϵ in all possible combina

Ex: 1)

$X \to aNbNa$
$N \to \epsilon$

$\Rightarrow$

$X \to aNbNa$
$X \to abNa$
$X \to aNba$
$X \to aba$

2)

$S \to aSa$
$S \to bSb$
$S \to \epsilon$

$\Rightarrow$

$S \to aSa$
$S \to bSb$
$S \to aa$
$S \to bb$

3)

$S \to a$
$S \to Xb$
$S \to aYa$
$X \to Y$
$X \to \epsilon$
$Y \to b$
$Y \to X$

X, Y: nullable.

$\Rightarrow$

$S \to a$
$S \to Xb$
$S \to aYa$
$X \to Y$
$Y \to b$
$Y \to X$
$S \to b$
$S \to aa$

2- Eliminate $A \to B$

Step1: if ($A \to B \in R$ or $A \overset{*}{\Rightarrow} B$)
and $B \to s_1$, $B \to s_2$, ..., $B \to s_n \in R$

then add $A \to s_1$, $A \to s_2$, ..., $A \to s_n$
(do not introduce $A \to C$!!)

do this for all pairs A, B simultaneously

Step2: Delete all $A \to B$

ex
$S \to A$
$S \to bb$
$A \to B$
$A \to b$
$B \to S$
$B \to a$

$\Rightarrow$

$S \to bb$, $A \to b$, $B \to a$ + $S \to b$, $S \to a$, $A \to a$, $A \to bb$, $B \to bb$, $B \to b$

$\left\{ \begin{array}{l} S \to A \\ A \to B, A \to b \end{array} \right.$ add $S \to b$

$\left\{ \begin{array}{l} S \Rightarrow A \Rightarrow B \\ B \to S, B \to a \end{array} \right.$ add $S \to a$

$\left\{ \begin{array}{l} A \to B \\ B \to S, B \to a \end{array} \right.$ add $A \to a$

$\left\{ \begin{array}{l} A \Rightarrow B \Rightarrow S \\ S \to A, S \to bb \end{array} \right.$ add $A \to bb$

$\left\{ \begin{array}{l} B \to S \\ S \to A, S \to bb \end{array} \right.$ add $B \to bb$

$\left\{ \begin{array}{l} B \Rightarrow S \Rightarrow A \\ A \to B, A \to b \end{array} \right.$ add $B \to b$

To test if $L(G) = \phi$:

Simply check ~~...~~ for all parse trees of height $\leq |V-\Sigma|$. If you find one than $L(G) \neq \phi$ otherwise $L(G) = \phi$.

Some Examples

① **Claim** $L = \{ a^i \mid i \text{ is prime} \}$ is not c.f.

Proof Let L be c.f. Then, by P.L., there is a k

Consider $\omega = a^m$, $m > k$, m = prime.

So, $\omega = \underbrace{a^p}_{u}\underbrace{a^q}_{v}\underbrace{a^r}_{x}\underbrace{a^s}_{y}\underbrace{a^t}_{z}$, $p+q+r+s+t = m$

$q+s \geq 1$

$q+r+s \leq k$

$r \geq 1$

$$a^p a^{nq} a^r a^{ns} a^t \in L$$

i.e. $p + nq + r + ns + t$ is prime

$p + n(q+s) + (r+t)$ is prime

let $n = (p + 2(q+s) + (r+t) + 2)$

$p + (p + 2(q+s) + (r+t) + 2)(q+s) + (r+t)$ is prime

$p + p(q+s) + 2(q+s)^2 + (r+t)(q+s) + 2(q+s) + (r+t)$ is prime

$p(q+s+1) + (r+t)(q+s+1) + (q+s)(2q+2s+2)$ is prime

$p(q+s+1) + (r+t)(q+s+1) + (q+s)\,2(q+s+1)$ is prime

$(q+s+1)(p + r + t + 2q + 2s)$ is prime

But $q+s+1 \geq 2$ and $p+r+t+2q+2s \geq 2$ $\Big\}$ $\because q+s \geq 1$

$\therefore$ we have a contradiction

$L = \{a^n b^{n^2} \mid n \geq 0\}$ is not C.F.

Let L be C.F. P.L. $\Rightarrow$ K.

Consider $z = a^k b^{k^2}$

$z = uvxyz$ $|vxy| \leq K$, $|vy| \geq 1$

① $\underline{v = a^l, y = a^m}$ $l+m \geq 1$

$uv^2 \cancel{x} y^2 z$: $\begin{cases} \# b's = K^2 \\ \# a's = K + \underline{l+m} > K \end{cases}$

↳ contradiction

② $\underline{v = b^l; y = b^m}$ $l+m \geq 1$

uv^2xy^2z : $\begin{cases} \# b's = K^2 + l+m \\ \underline{\# a's = K} \end{cases}$

↳ Contradiction

③ $\underline{v = a^l, y = b^m}$ $1 \leq l \leq K$, $1 \leq m \leq K$

uv^2xy^2z : $a^{K+l} b^{K^2+m} \in L$

$K^2 + m = (K+l)^2$

$K^2 + m = K^2 + l^2 + 2Kl$

$\cancel{K^2} + m = \cancel{K^2} + l^2 + 2Kl$

$\underbrace{m}_{\leq K} = \underbrace{l^2 + 2Kl}_{> K}$

Contradiction.

④ $\underline{v, y \text{ contain}}$ $\underline{\text{a's and b's}}$

On pumping we get a's after b's

Contradiction