#1)

Let $w = a_1 a_2 \ldots a_n$. Since there can be at most one rule with S on the left, and starting with a_1 on the right, the derivation must begin $S \Rightarrow a_1 A A_2 \ldots A_m$. Next, substitute for A_1 and for each A_1 there is at most one choice.. ∴ $S \stackrel{*}{\Rightarrow} a_1 a_2 B_1 B_2 \ldots A_2 \ldots A_m$. Since we are substituting at most one character at a time, by the time the entire string is processed, the length can be atmost $|w|$. At each step, only one terminal symbol is produced and the derivation is completed in no more than $|w|$ steps. □

2) Given a CFG if we eliminate all its useless productions we still get a smaller CFG with the same language.

To determine the useful variables and productions we can start with V_1 = empty set. Then repeat the following until there are no more variables added to V_1: For each production $A \to X_1 \cdots X_n$, with all X_i's that are variables in V_1, and A to V_1. if the start variable is not in V_1 then we know the language is empty. So we can delete all productions.

Otherwise, if S is in V_1, it still might not be the case that every variable in V_1 is useful, so we ~~get~~ set $V_2 = \{S\}$. then repeat the following until there are no more variables added to V_2: For each production $A \to X_1 \cdots X_n$, with all X_i's that are variables ~~added~~ in V_1 and with A added to V_2, add each variable on the left hand side to V_2, After this procedure terminates, take V_2 to be the set of useful variables. All other variables and production they are involved in are useless.

3

GRAMMAR IN

CNF, BUT NOT IN GNF

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

GNF MUST HAVE THE FORM

$A \to aB \ldots$

WHERE a IS A TERMINAL AND B IS A SEQUENCE OF VARIABLES

GRAMMAR IN

GNF BUT NOT IN CNF

$A \to aB$
$A \to a$
$B \to bAAA$

CNF CAN ONLY HAVE RULES WHERE

$A \to BC$

OR

$A \to a$

(1)

#4

$A \rightarrow AB \mid a$ Is in normal form.
$B \rightarrow AA$

Is "aaaaa" in the grammar?

<u>a</u>aaaa $V_{11} = \{A\}$ a<u>a</u>aaa $V_{22} = \{A\}$ aa<u>a</u>aa $V_{33} = \{A\}$ aaa<u>a</u>a $V_{44} = \{A\}$ aaaa<u>a</u> $V_{55} = \{A\}$

<u>aa</u>aaa
split: a a
$V_{11} = \{A\}$ $V_{22} = \{A\}$ => AA
$V_{12} = \{B\}$

a<u>aa</u>aa
split: a a
$V_{22} = \{A\}$ $V_{33} = \{A\}$ => AA
$V_{23} = \{B\}$

aa<u>aa</u>a
split: a a
$V_{33} = \{A\}$ $V_{44} = \{A\}$
=> AA
$V_{34} = \{B\}$

aaa<u>aa</u>
split: a a
$V_{44} = \{A\}$ $V_{55} = \{A\}$ => AA
$V_{45} = \{B\}$

<u>aaa</u>aa
split: a aa
$V_{11} = \{A\}$ $V_{23} = \{B\}$
=> AB
$V_{13} = \{A\}$ ✓

check: aa a
$V_{12} = \{B\}$ $V_{33} = \{A\}$
=> BA
Not in the grammar

a<u>aaa</u>a
split: a aa
$V_{22} = \{A\}$ $V_{34} = \{B\}$
=> AB
$V_{24} = \{A\}$ ✓

check: aa a
$V_{23} = \{B\}$ $V_{44} = \{A\}$
=> BA
Not in the grammar

$V_{35} = \{A\}$ same process

#4 continued

aaaaa

split: a aaa

$V_{11} = \{A\}$ $V_{24} = \{A\}$

$\Rightarrow AA$

$V_{14} = \{B\}$ ✓

check: aa aa

$V_{12} = \{B\}$ $V_{34} = \{B\}$

$\Rightarrow BB$

Not in the grammar

same

check: aaa a

$V_{13} = \{A\}$ $V_{44} = \{A\}$

$\Rightarrow AA$

$V_{14} = \{B\}$

aaaaa

split: a aaa

$V_{22} = \{A\}$ $V_{35} = \{A\}$

$\Rightarrow AA$

$V_{25} = \{B\}$ ✓

check: aa aa

$V_{23} = \{B\}$ $V_{45} = \{B\}$

$\Rightarrow BB$

Not in the grammar

check: aaa a

$V_{24} = \{A\}$ $V_{55} = \{A\}$

$\Rightarrow AA$

$V_{25} = \{B\}$

aaaaa

split: a aaaa

$V_{11} = \{A\}$ $V_{25} = \{B\}$

$\Rightarrow AB$

$V_{15} = \{B\}$.

check: aa aaa

$V_{12} = \{B\}$ $V_{35} = \{A\}$

$\Rightarrow BA$

Not in grammar

2 4 1

3 2

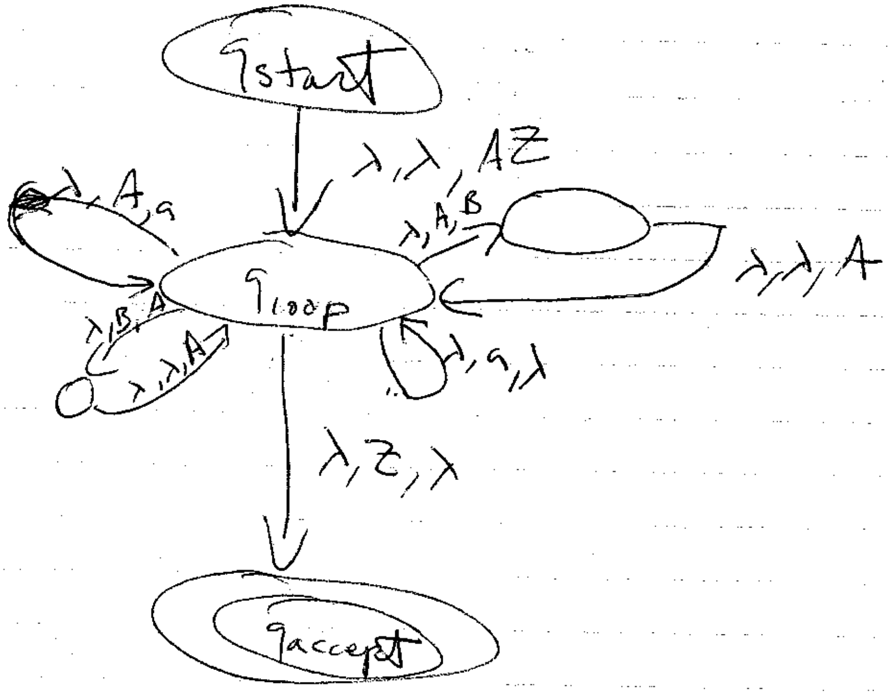
check: aaa aa

$V_{13} = \{A\}$ $V_{45} = \{B\}$

$\Rightarrow AB$

$V_{15} = \{A\}$ ✓ pick this one because its the start. ∴ aaaaa is in the language.

5



6 Definition: DCFL

A language L is said to be a DCFL iff there exists a dpda M such that $L = L(M)$.

Prove that $\{a^n b^n \mid n \geq 0\}$ is DCFL.

$M = (Q, \Sigma, \Gamma, \delta, q_0, z, F)$

$Q = \{q_0, q_1, q_2\}$

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

$\Gamma = \{0, 1\}$

$z = 0$

$F = \{q_0\}$

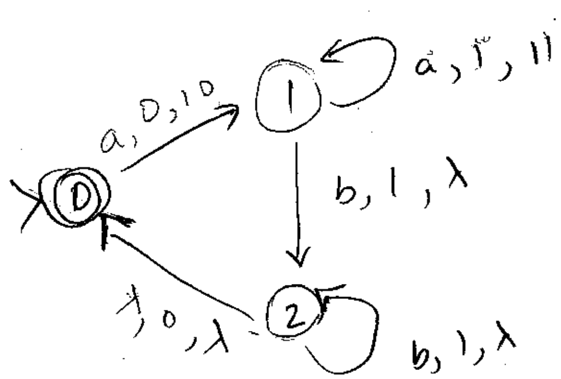
$\delta(q_0, a, 0) = \{(q_1, 10)\}$

$\delta(q_1, a, 1) = \{(q_1, 11)\}$

$\delta(q_1, b, 1) = \{(q_2, \lambda)\}$

$\delta(q_2, b, 1) = \{(q_2, \lambda)\}$

$\delta(q_2, \lambda, 0) = \{(q_0, \lambda)\}$



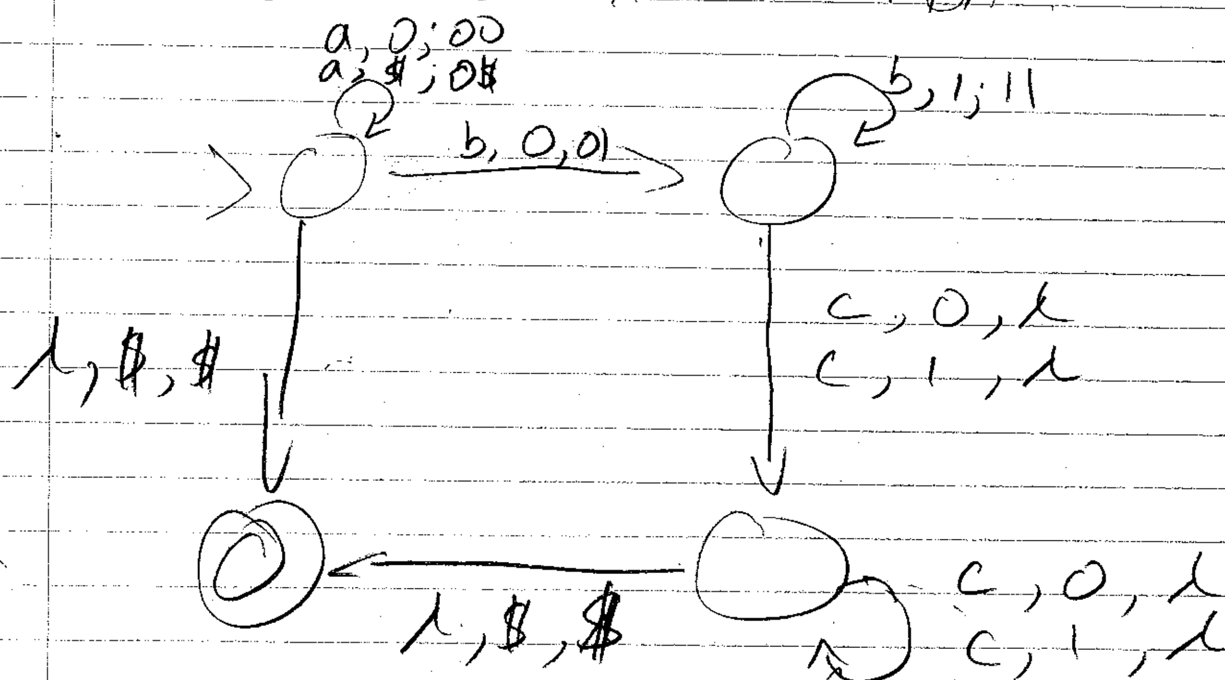
#7

Because Griebach Normal Form requires a terminal followed by a set of variables. However, in the book, the initial terminal may be lambda istead.

8. first part

$\{a^n b^n c^{n+m} \mid n, m \geq 0\}$ is a DCFL

because there ~~is~~ exists a PDA:



8.b prove $L = \{ a^n b^n c^{nm} \mid n, m \geq 0 \}$ is not context-free

Let PDA M recognize L
let p be the pumping length of M

String S is in L.

$S = a^p b^p c^{pp}$, where $|s| > p$, and $|vxy| \leq p$, $|vy| > 0$

let $h+k+j \leq p$, $h, j > 0$

$S = uv^ixy^iz$, where $i \geq 0$

Case 1 vxy covers only a single ~~letter~~ letter type

① $vxy = a^m$ ← where $m \leq p$

$u = \{\}$ $v = a^h$ $x = a^k$ $y = a^j$ $z = b^p c^{pp}$

when $i = 0$:

$uv^0xy^0z = a^k b^p c^{pp}$ // since $k < p$, $k \cdot p \neq pp$ not in L

② $vxy = b^m$

$u = a^p$ $v = b^h$ $x = b^k$ $y = b^j$ $z = c^{pp}$

when $i = 0$:

$uv^0xy^0z = a^p b^k c^{pp}$ // ditto

b & c case similar

③ $vxy = c^{(h+k+j)}$

$u = a^p b^p$ $v = b^h$ $x = b^k$ $y = b^j$ $z = c^{pp-(h+k+j)}$

when $i = 0$:

$uv^0xy^0z = a^p b^p c^k c^{pp-h-k-j} = a^p b^p c^{pp-h-j}$ // $p \cdot p \neq pp - h - j$, $h, j > 0$ not in L

Two letter case can't happen because of pump up will get letters out of order

Case 2 con't

I will post student generated .Jff file

#10 Turing Machine M accepts w if a there is a sequence of configurations $C_1, C_2 \ldots C_k$ such that C_1 is start config of M on w ; for i between 1 & $K-1$, C_i yields C_{i+1} & C_k is an accept config.

added definition of yield from book