TOC Assignment - 3

Solutions

Problem 1. Find a CFG for the language over $\{0, 1\}$ consisting of those strings in which the ratio of the number of 1's to the number of 0's is three to two. Also, design a PDA.

Solution. Given,

$$\Sigma = \{0, 1\}^*$$
$$let, x \in L \; is \; a \; CFG$$
$$2n_1(x) = 3n_0(x)$$
$$\frac{n_1(x)}{n_0(x)} = \frac{3}{2}$$

simplest string would be 2 0's and 3 1's

$$\frac{5!}{2!.3!} = 10 \; simple \; strings$$

let's take one string and right its permutations later 00111 for S generating 1's. $S \to AABBB$ for every A there is 0 and for every B there is 1
All 10 combinations of 2'As and 3'Bs are
$AABBB, BAABB, BBAAB, BBBAA, ABBBA, BABAB, ABABB, BBABA, BABBA, ABBAB$

$$A \to 0S|S0|0$$
$$B \to 1S|S1|1$$

this is already in a simplified form
lets convert in PDA
Start :

$$\delta(q_0, \epsilon, \epsilon) = (q_0, z_0)$$
$$\delta(q_0, \epsilon, z_0) = (q_1, Sz_0)$$

Non deterministic push states :

$$\delta(q_1, \epsilon, S) = (q_1, AABBBz_0) = (q_1, BBBAAz_0)$$
$$\delta(q_1, \epsilon, S) = (q_1, BAABBz_0) = (q_1, ABABBz_0)$$
$$\delta(q_1, \epsilon, S) = (q_1, BBAABz_0) = (q_1, BBABAz_0)$$
$$\delta(q_1, \epsilon, S) = (q_1, ABBABz_0) = (q_1, BABBAz_0)$$
$$\delta(q_1, \epsilon, S) = (q_1, BABABz_0) = (q_1, ABBBAz_0)$$

pop :

$$\begin{aligned}
&\delta(q_1, \epsilon, A) = (q_1, 0S) = (q_1, S0) = (q_1, 0) \\
&\delta(q_1, \epsilon, B) = (q_1, 1S) = (q_1, S1) = (q_1, 1) \\
&\delta(q_1, 1, 1) = (q_1, \epsilon) \\
&\delta(q_1, 0, 0) = (q_1, \epsilon) \\
&\delta(q_1, \epsilon, z_0) = (q_f, z_0)
\end{aligned}$$

Non deterministic rejected states :

$$\begin{aligned}
&\delta(q_1, 1, 0) = \delta(q_1, 1, z_0) = (q_r, \epsilon) \\
&\delta(q_1, 0, 1) = \delta(q_1, o, z_0) = (q_r, \epsilon)
\end{aligned}$$

Question 2.
Convert the following to CNF.

$$\text{S} \to \text{ABA, A} \to \text{aA, A} \to \epsilon\text{, B} \to \text{bB, B} \to \epsilon.$$

Solution.
The given CFG is

$$\begin{aligned}
&S \to ABA \\
&A \to aA \\
&A \to \epsilon \\
&B \to bB \\
&B \to \epsilon
\end{aligned}$$

We now Simplify the CFG using the following rules in order

(i) Eliminate ϵ productions

(ii) Eliminate unit productions

(iii) Eliminate useless symbols

Now using (i), we get

$$\begin{aligned}
&S \to ABA \mid BA \mid AB \mid AA \mid A \mid B \mid \epsilon. \\
&A \to aA \mid a \\
&B \to bB \mid b
\end{aligned}$$

Now using (ii), we get

$$\begin{aligned}
&S \to ABA \mid BA \mid AB \mid AA \mid aA \mid a \mid bB \mid b \mid \epsilon. \\
&A \to aA \mid a \\
&B \to bB \mid b
\end{aligned}$$

As there're no useless symbols, the CFG can't be simplified any further.
Since S-> epsilon is present, CNF does not exist.

Question 3.
Is the following grammar Ambiguous.

$$\text{S} \to \text{Sb, S} \to \text{aSb, S} \to \text{Sa, S} \to \text{a.}$$

Solution.
A context free grammar is called ambiguous if there exists more than one Left most derivation or more than one Right most derivation for a string which is generated by grammar.
Now, Let's prove that the grammar is ambiguous by taking an example. We take the string **aab** as an example.
First way is to use the production S $\to$ Sb, then use the production S $\to$ Sa and now terminate using the production S $\to$ a. This can be written as

$$\begin{aligned}
S &\to Sb \\
&\to Sab \\
&\to aab
\end{aligned}$$

Second way is to use the production S $\rightarrow$ aSb, then terminate using the production S $\rightarrow$ a. This can be written as

$$\begin{aligned} S &\rightarrow aSb \\ &\rightarrow aab \end{aligned}$$

As there exists more than one Left most derivation for the string **aab**, The given Grammar is ambiguous. □

Question 4.
What is L(G), S $\rightarrow$ aS | aSbS | ϵ. Is G ambiguous. If so, find an equivalent unambiguous grammar. Also, design a PDA.

Solution.
L(G) is the strings over {a,b} such that $n(a) \geq n(b) \geq 0$.
As said above, a grammar is ambiguous if there exists more than one LMD or more than one RMD for a string which is generated by grammar. The Given CFG is Ambiguous which can be proved using an example using the string **aab**.

$$\begin{aligned} S &\rightarrow aS \\ &\rightarrow aaSbS \\ &\rightarrow aabS \\ &\rightarrow aab \end{aligned} \qquad\qquad \begin{aligned} S &\rightarrow aSbS \\ &\rightarrow aaSbS \\ &\rightarrow aabS \\ &\rightarrow aab \end{aligned}$$

As there are more than one Left most derivation for the string **aab**, the given Grammar G is ambiguous.
Equivalent Unambiguous grammar :

S $\rightarrow$ aS | XS | ϵ

X $\rightarrow$ aXb | bXa | ϵ

PDA for the Grammar:
Pushing A's when A's are present on the stack when we encounter a :

$$\boldsymbol{\delta(q_0, a, z_0) = (q_0, A, z_0)}$$

$$\boldsymbol{\delta(q_0, a, A) = (q_0, A, A)}$$

Pushing B's when B's are present on the stack when we encounter b :

$$\boldsymbol{\delta(q_0, b, z_0) = (q_0, B, z_0)}$$

$$\boldsymbol{\delta(q_0, b, B) = (q_0, B, B)}$$

Pop upon encountering a different symbol :

$$\boldsymbol{\delta(q_0, a, B) = (q_0, \epsilon)}$$

$$\boldsymbol{\delta(q_0, b, A) = (q_0, \epsilon)}$$

Reject when $n(b) \geq n(a)$:

$$\boldsymbol{\delta(q_0, \epsilon, B) = (q_r, z_0)}$$

Accept the string otherwise :

$$\boldsymbol{\delta(q_0, \epsilon, A) = (q_f, z_0)}$$
$$\boldsymbol{\delta(q_0, \epsilon, z_0) = (q_f, z_0)}$$

Question 5.
Find PDA, CFG; L = $\{a^m b^n \mid m \leq n \leq 2m\}$.

Solution.
CFG for the given Language L = $\{a^m b^n \mid m \leq n \leq 2m\}$ is

$$\boldsymbol{S \rightarrow aSb \mid aSbb \mid \epsilon}$$

PDA for the Language :

$$\delta(q_0, \epsilon, z_0) = (q_1, S) \qquad \text{Insert the Start Symbol}$$

$$\left.\begin{array}{l} \delta(q_1, \epsilon, S) = (q_1, aSb) \\ \delta(q_1, \epsilon, S) = (q_1, aSbb) \\ \delta(q_1, \epsilon, S) = (q_1, \epsilon) \end{array}\right\} \text{ Generation Rules}$$

$$\left.\begin{array}{l} \delta(q_1, a, a) = (q_1, \epsilon) \\ \delta(q_1, b, b) = (q_1, \epsilon) \end{array}\right\} \text{ Pop Operations}$$

$$\left.\delta(q_1, \epsilon, z_0) = (q_f, z_0)\ \right\} \text{ Accepts String}$$

$$\left.\begin{array}{l} \delta(q_1, a, b) = (q_r, z_0) \\ \delta(q_1, b, a) = (q_r, z_0) \end{array}\right\} \text{ Rejects String}$$

Question 6.
Show that the language a^p , p is prime, is not context free.

Solution.
A language is not context-free if (Pumping Lemma)

$$\forall n \exists u (|u| \geq n \wedge \forall v \forall w \forall x \forall y \forall z (u = vwxyz, |wy| \geq 1\ |wxy| \leq n \rightarrow \exists i (i \geq 0 \wedge vw^i xy^i z \notin L)))$$

We take prime number $p \geq n \implies a^p \in L$

$$a^p = a^b a^j a^k a^l a^m \qquad p = b + j + k + l + m$$

$$v = a^b \qquad w = a^j \qquad x = a^k \qquad y = a^l \qquad z = a^m$$

$$\text{So, } j+l \geq 1 \qquad \text{and} \qquad j+k+l \leq n$$

We can choose $i = p+1$ and let the string generated be S

$$\begin{aligned}\implies vw^ixy^iz = a^b a^{j(p+1)} a^k a^{l(p+1)} a^m &= a^{b+k+m} a^{(j+l)(p+1)} \\ &= a^{p-j-l} a^{(j+l)(p+1)} \\ &= a^{p+(j+l)(p)} \\ &= a^{p(j+l+1)}\end{aligned}$$

As $j+l+1 \geq 2 \implies p(j+l+1)$ is not prime
$\Rightarrow S \notin L$. Therefore, L **is not context free**.

Question 7.
Is L $= \{a^n b^m c^k \mid n, m, k \geq 1, 2n = 3k \text{ or } 5n = 7m\}$ context free.

Solution.
Yes, the given Language is context free. We prove this by designing a CFG for the given Language L. We can do this by first designing CFG for $2n = 3k$ and then designing for $5n = 7m$ then taking the union of CFG's of both.

$$L_1 = \{a^n b^m c^k \mid n, m, k \geq 1, 2n = 3k\}$$

CFG for the Language L_1 is

$S_1 \to aaaBcc \mid aaaS_1cc$	For generating 3 a's for every 2 c's
$B \to bB \mid b$	For generating how many ever b's needed

$$L_2 = \{a^n b^m c^k \mid n, m, k \geq 1, 5n = 7m\}$$

CFG for the Language L_2 is

$S_2 \to AC$	
$C \to c\ C \mid c$	For generating how many ever c's needed
$A \to aaaaaaaAbbbbb \mid aaaaaaabbbbb$	For generating 7 a's for every 5 b's

$$L = \{a^n b^m c^k \mid n, m, k \geq 1, 2n = 3k \text{ or } 5n = 7m\}$$

Now, CFG for the given Language L is

$$\begin{aligned}&\boldsymbol{S \to S_1 \mid S_2}\\ &\boldsymbol{S_1 \to aaaBcc \mid aaaS_1cc}\\ &\boldsymbol{B \to bB \mid b}\\ &\boldsymbol{S_2 \to AC}\\ &\boldsymbol{C \to c\ C \mid c}\\ &\boldsymbol{A \to aaaaaaaAbbbbb \mid aaaaaaabbbbb}\end{aligned}$$

Question 8.
For the language of equal no of 0's and 1's, write the Greibach Normal Form.

Solution.
Simplified CFG for the Language of equal no of 0's and 1's :

$$S \to SS \mid 0S1 \mid 1S0 \mid 01 \mid 10$$
$$A_6 \to A_6A_6 \mid 0A_61 \mid 1A_60 \mid 01 \mid 10$$

(i) A $\to$ BC where A,B,C are non-terminals

(ii) A $\to$ a where a $\in \Sigma$

Converting the above CFG to CNF by following the above rules

$$\begin{aligned} &A_6 \to A_6A_6 \mid A_2A_3 \mid A_4A_5 \mid A_2A_4 \mid A_4A_2 \\ &A_2 \to 0 \\ &A_3 \to A_6A_4 \\ &A_4 \to A_1 \\ &A_5 \to A_6A_2 \end{aligned}$$

Now, Let's convert CNF to GNF using the rules :

(i) If $A_k \to A_j\alpha$, $j < k$, then for all $A_i \to \beta$
ADD $A_k \to \beta\alpha$
REMOVE $A_k \to A_j\alpha$

(ii) If $A_k \to A_k\alpha$
ADD $B_k \to \alpha$ and $B_k \to \alpha B_k$
REMOVE $A_k \to A_k\alpha$

(iii) If $A_k \to \beta$ (β doesn't brgin with A_k)
ADD $A_k \to \beta B_k$

Now Applying the above rules

$$\left.\begin{array}{ll} \textbf{REMOVE} & A_6 \to A_2A_3 \\ \textbf{ADD} & A_6 \to 0A_3 \\ \textbf{REMOVE} & A_6 \to A_4A_5 \\ & A_6 \to A_2A_4 \\ & A_6 \to A_4A_2 \\ \textbf{ADD} & A_6 \to 1A_5 \\ & A_6 \to 0A_4 \\ & A_6 \to 1A_2 \end{array}\right\} \text{Rule 1}$$

$$\left.\begin{array}{ll}\textbf{REMOVE} & A_6 \rightarrow A_2A_3 \\ \textbf{ADD} & B_6 \rightarrow A_6 \\ & B_6 \rightarrow A_6B_6\end{array}\right\} \text{ Rule 2}$$

$$\left.\begin{array}{ll}\textbf{ADD} & A_2 \rightarrow 0B_2 \\ & A_4 \rightarrow 1B_4 \\ & A_6 \rightarrow 0A_3A_6 \\ & A_6 \rightarrow 1A_5A_6 \\ & A_6 \rightarrow 1A_5A_6 \\ & A_6 \rightarrow 0A_4A_6 \\ & A_6 \rightarrow 1A_2A_6\end{array}\right\} \text{ Rule 3}$$

Now, we have

$$\begin{aligned}
&A_6 \rightarrow 0A_3 \mid 1A_5 \mid 0A_4 \mid 1A_2 \mid 0A_3B_6 \mid 1A_5B_6 \mid 0A_4B_6 \mid 1A_2B_6 \\
&A_2 \rightarrow 0 \mid 0B_2 \\
&A_3 \rightarrow A_6A_4 \\
&A_4 \rightarrow 1 \mid 1B_4 \\
&A_5 \rightarrow A_6A_2 \\
&B_6 \rightarrow A_6B_6
\end{aligned}$$

Now substituting A_6 in A_3, A_5 and B_6 to convert the above to GNF

$$\begin{aligned}
&A_6 \rightarrow 0A_3 \mid 1A_5 \mid 0A_4 \mid 1A_2 \mid 0A_3B_6 \mid 1A_5B_6 \mid 0A_4B_6 \mid 1A_2B_6 \\
&A_2 \rightarrow 0 \mid 0B_2 \\
&A_3 \rightarrow 0A_3A_4 \mid 1A_5A_4 \mid 0A_4A_4 \mid 1A_2A_4 \mid 0A_3B_6A_4 \mid 1A_5B_6A_4 \mid 0A_4B_6A_4 \mid 1A_2B_6A_4 \\
&A_4 \rightarrow 1 \mid 1B_4 \\
&A_5 \rightarrow 0A_3A_2 \mid 1A_5A_2 \mid 0A_4A_2 \mid 1A_2A_2 \mid 0A_3B_6A_2 \mid 1A_5B_6A_2 \mid 0A_4B_6A_2 \mid 1A_2B_6A_2 \\
&B_6 \rightarrow 0A_3B_6 \mid 1A_5B_6 \mid 0A_4B_6 \mid 1A_2B_6 \mid 0A_3B_6B_6 \mid 1A_5B_6B_6 \mid 0A_4B_6B_6 \mid 1A_2B_6B_6
\end{aligned}$$

As B_2 and B_4 are useless symbols and no state can be reached using them, we remove it from the above transitions. Therefore, The Final GNF for this CFG is

$$\boldsymbol{\begin{aligned}
&A_6 \rightarrow 0A_3 \mid 1A_5 \mid 0A_4 \mid 1A_2 \mid 0A_3B_6 \mid 1A_5B_6 \mid 0A_4B_6 \mid 1A_2B_6 \\
&A_2 \rightarrow 0 \\
&A_3 \rightarrow 0A_3A_4 \mid 1A_5A_4 \mid 0A_4A_4 \mid 1A_2A_4 \mid 0A_3B_6A_4 \mid 1A_5B_6A_4 \mid 0A_4B_6A_4 \mid 1A_2B_6A_4 \\
&A_4 \rightarrow 1 \\
&A_5 \rightarrow 0A_3A_2 \mid 1A_5A_2 \mid 0A_4A_2 \mid 1A_2A_2 \mid 0A_3B_6A_2 \mid 1A_5B_6A_2 \mid 0A_4B_6A_2 \mid 1A_2B_6A_2 \\
&B_6 \rightarrow 0A_3B_6 \mid 1A_5B_6 \mid 0A_4B_6 \mid 1A_2B_6 \mid 0A_3B_6B_6 \mid 1A_5B_6B_6 \mid 0A_4B_6B_6 \mid 1A_2B_6B_6
\end{aligned}}$$

□