COMP 335: Introduction to Theoretical Computer Science Fall 2014

Solutions to Assignment 3

1. Give a CFG that generates all regular expressions on the alphabet $\Sigma = \{a, b\}$. Clearly specify the alphabet of your CFG, its terminals, variables, and productions.

   Ans:

   $G = \langle V, T, R, P \rangle$ where

   $V = \{R, I\}$,
   $T = \Sigma \cup \{(,), ^\star, +\}$,
   $P = \{R \rightarrow R + R \mid RR \mid R^\star \mid (R) \mid I,$
   $\quad I \rightarrow a \mid b\}$.

2. Let L be the language of valid arithmetic expressions using $+$, $-$, $*$, $/$, (,), and ˆ and the variables $\{a, b, c\}$.

   (a) Give an unambiguous CFG that generates L.

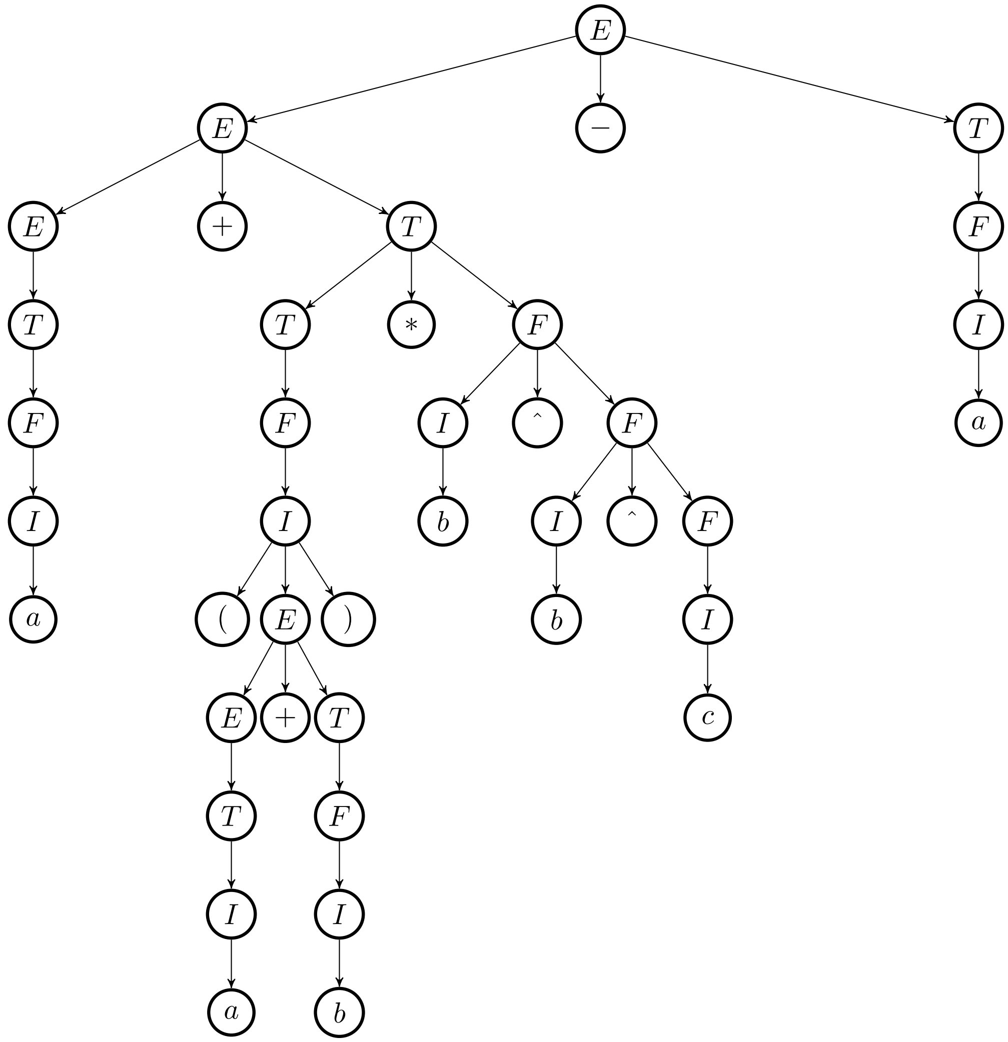
   Ans: Notice that the precedence of the ˆ operator is from right-to-left.

   $G = \langle V, T, E, P \rangle$ where
   $V = \{E, T, F, I\}$,
   $T = \{a, b, c\} \cup \{+, -, *, /, \hat{}, (,)\}$,
   $P = \{E \rightarrow E + T \mid E - T \mid T,$
   $\quad T \rightarrow T * F \mid T/F \mid F,$
   $\quad F \rightarrow I\hat{}F \mid I,$
   $\quad I \rightarrow a \mid b \mid c \mid (E)\}$.

   Note that the above grammar does not generate strings with unary operators. Solutions that allow unary operators will also be accepted.

   (b) Give the derivation tree for the expression $a + (a + b) * b\hat{}b\hat{}c - a$.

   Ans:



3. Give a CFG for each of the following languages:

   (a) $L = \{a^n b^m c^m d^k : n > 0, m > 0, k = 2n\}$
   Ans: $S \to aSdd \mid aAdd$
   $A \to bAc \mid bc$

   (b) $L = \{a^n b^m c^k : n + m \geq k \geq 0\}$
   Ans: $S \to aSc \mid aS \mid A$
   $A \to bAc \mid bA \mid \lambda$

   (c) $L = \{a^n b^m c^k : n + k \leq m\}$
   Ans: $S \to ABC$
   $A \to aAb \mid \lambda$
   $B \to bB \mid \lambda$
   $C \to bCc \mid \lambda$

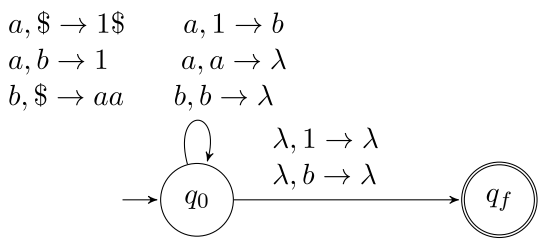
4. For each of the following languages, design a PDA that accepts the language: Note: In all of the solutions below, \$ is the bottom of stack marker. z can also be used as the bottom of stack marker.

   (a) $L = \{w \in \{a, b\}^\star : n_a(w) > 2n_b(w)\}$

Ans: We use the following stack symbol indications:

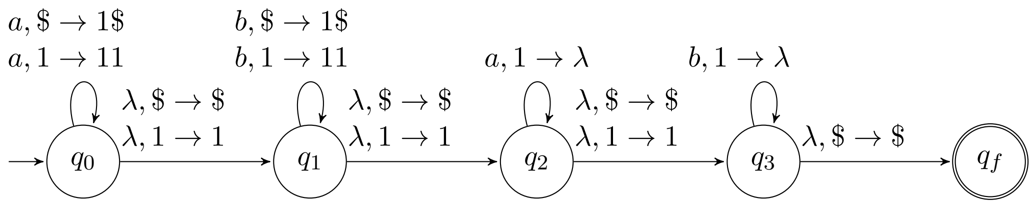
\$: $n_a(w) = 2n_b(w)$	1: 1a ahead
b: 2a's ahead	a: 1a behind

The PDA accepting L is, as follows:



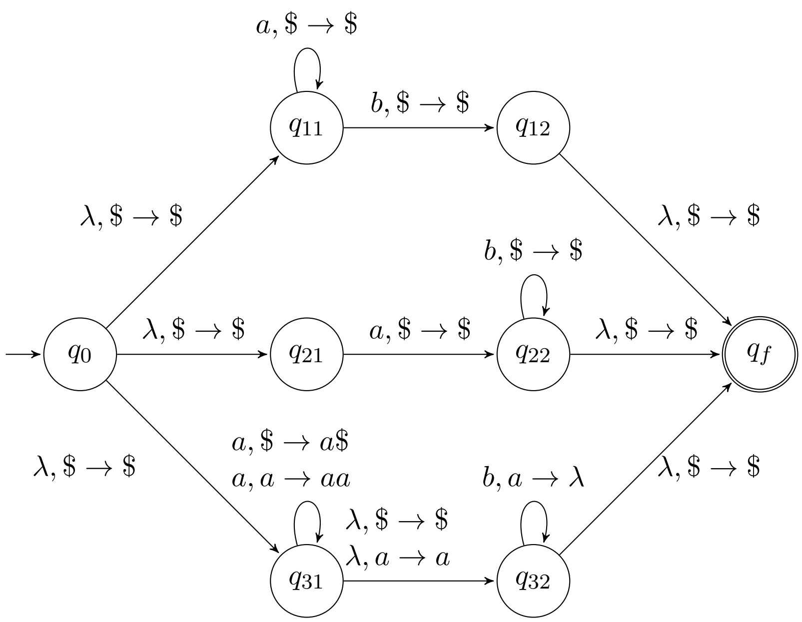
(b) $L = \{a^h b^k a^m b^n : h + k = m + n\}$

Ans:



(c) $L = \{a^\star b\} \cup \{ab^\star\} \cup \{a^i b^i \mid i \geq 0\}$

Ans:



5. Consider the following grammar:

$S \to AB \mid aaB$
$A \to Aa \mid a$
$B \to b$

(a) Show that G is ambiguous.

Ans: The string aab has two distinct leftmost derivations as shown below. Each of these correspond to different derviation trees.

$S \Rightarrow aaB \Rightarrow aab$

$S \Rightarrow AB \Rightarrow AaB \Rightarrow aaB \Rightarrow aab$

(b) Obtain $L(G)$.

Ans: $L(G) = a^\star ab$.

(c) Provide an unambiguous grammar G' that generates $L(G)$.

Ans: By removing the redundant production $S \to aaB$, we obtain the following unambiguous grammar:

$S \to AB,\ A \to Aa \mid a,\ B \to b.$

Another unambiguous grammar for the language a^*ab is:

$S \to aS \mid ab.$

6. Consider the following grammar:

$S \to ABAC \mid BaA$
$A \to Aa \mid BAbC \mid \lambda$
$B \to bB \mid aBaC \mid \lambda$
$C \to a \mid b,$

(a) Convert G into Chomsky normal form.

Ans: We apply the procedure to convert G to CNF. First we remove the λ-transitions.

$S \to a \mid aA \mid Ba \mid BaA \mid C \mid AC \mid BC \mid AAC \mid ABC \mid BAC \mid ABAC,$
$A \to a \mid Aa \mid bC \mid AbC \mid BbC \mid BAbC,$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

Next we remove unit productions:

$S \to a \mid b \mid aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to Ba \mid BaA \mid BC \mid BAC,$
$A \to a \mid bC,$
$A \to Aa \mid AbC,$
$A \to BbC \mid BAbC,$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

Next we introduce new variables T_a and T_b:

$S \to a \mid b \mid T_aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to BT_a \mid BT_aA \mid BC \mid BAC,$
$A \to a \mid T_bC,$
$A \to AT_a \mid AT_bC,$
$A \to BT_bC \mid BAT_bC,$
$B \to b \mid T_bB \mid T_aT_aC \mid T_aBT_aC,$
$C \to a \mid b,$
$T_a \to a, T_b \to b$

Finally, we convert all productions to CNF:

$S \to a \mid b \mid T_a A,$
$S \to AC \mid AV_1 \mid AV_2 \mid AV_3,$
$S \to BT_a \mid BV_4 \mid BC \mid BV_1,$
$A \to a \mid T_b C,$
$A \to AT_a \mid AV_5,$
$A \to BV_5 \mid BV_6,$
$B \to b \mid T_b B \mid T_a V_7 \mid T_a V_8,$
$C \to a \mid b,$
$T_a \to a, T_b \to b,$
$V_1 \to AC, V_2 \to BC, V_3 \to BV_1, V_4 \to T_a A,$
$V_5 \to T_b C, V_6 \to AV_5, V_7 \to T_a C, V_8 \to BV_7$

(b) Convert G into Greibach normal form.

Ans: Students are not required to do this part, and will not be graded on it, but here is the answer in any case. We can apply the conversion procedure to GNF on G. Take the simplified grammar in the third step in 6a.

$S \to a \mid b \mid aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to Ba \mid BaA \mid BAC,$
$A \to a \mid bC,$
$A \to Aa \mid AbC,$
$A \to BbC \mid BAbC,$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

By removing the left recursion on A, a non recursive grammar is obtained, as follows:

$S \to a \mid b \mid aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to Ba \mid BaA \mid BAC,$
$A \to aA' \mid bCA',$
$A \to BbCA' \mid BAbCA',$
$A' \to aA' \mid bCA' \mid \lambda,$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

By removing $A' \to \lambda$, we get:

$S \to a \mid b \mid aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to Ba \mid BaA \mid BAC,$
$A \to a \mid aA' \mid bC \mid bCA',$
$A \to BbC \mid BbCA' \mid BAbC \mid BAbCA',$
$A' \to a \mid aA' \mid bC \mid bCA',$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

By replacing the left-most B's, we get:

$S \to a \mid b \mid aA,$
$S \to AC \mid AAC \mid ABC \mid ABAC,$
$S \to ba \mid bBa \mid aaCa \mid aBaCa,$
$S \to baA \mid bBaA \mid aaCaA \mid aBaCaA,$
$S \to bAC \mid bBAC \mid aaCAC \mid aBaCAC,$
$A \to a \mid aA' \mid bC \mid bCA',$
$A \to bbC \mid bBbC \mid aaCbC \mid aBaCbC,$
$A \to bbCA' \mid bBbCA' \mid aaCbCA' \mid aBaCbCA',$
$A \to bAbC \mid bBAbC \mid aaCAbC \mid aBaCAbC,$
$A \to bAbCA' \mid bBAbCA' \mid aaCAbCA' \mid aBaCAbCA',$
$A' \to a \mid aA' \mid bC \mid bCA',$
$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

Similarly, by resolving leftmost A's, we get:

$S \to a \mid b \mid aA,$
$S \to aC \mid aA'C \mid bCC \mid bCA'C,$
$S \to bbCC \mid bBbCC \mid aaCbCC \mid aBaCbCC,$
$S \to bbCA'C \mid bBbCA'C \mid aaCbCA'C \mid aBaCbCA'C,$
$S \to bAbCC \mid bBAbCC \mid aaCAbCC \mid aBaCAbCC,$
$S \to bAbCA'C \mid bBAbCA'C \mid aaCAbCA'C \mid aBaCAbCA'C,$
$S \to aAC \mid aA'AC \mid bCAC \mid bCA'AC,$
$S \to bbCAC \mid bBbCAC \mid aaCbCAC \mid aBaCbCAC,$
$S \to bbCA'AC \mid bBbCA'AC \mid aaCbCA'AC \mid aBaCbCA'AC,$
$S \to bAbCAC \mid bBAbCAC \mid aaCAbCAC \mid aBaCAbCAC,$
$S \to bAbCA'AC \mid bBAbCA'AC \mid aaCAbCA'AC \mid aBaCAbCA'AC,$
$S \to aBC \mid aA'BC \mid bCBC \mid bCA'BC,$
$S \to bbCBC \mid bBbCBC \mid aaCbCBC \mid aBaCbCBC,$
$S \to bbCA'BC \mid bBbCA'BC \mid aaCbCA'BC \mid aBaCbCA'BC,$
$S \to bAbCBC \mid bBAbCBC \mid aaCAbCBC \mid aBaCAbCBC,$
$S \to bAbCA'BC \mid bBAbCA'BC \mid aaCAbCA'BC \mid aBaCAbCA'BC,$
$S \to aBAC \mid aA'BAC \mid bCBAC \mid bCA'BAC,$
$S \to bbCBAC \mid bBbCBAC \mid aaCbCBAC \mid aBaCbCBAC,$
$S \to bbCA'BAC \mid bBbCA'BAC \mid aaCbCA'BAC \mid aBaCbCA'BAC,$
$S \to bAbCBAC \mid bBAbCBAC \mid aaCAbCBAC \mid aBaCAbCBAC,$
$S \to bAbCA'BAC \mid bBAbCA'BAC \mid aaCAbCA'BAC \mid aBaCAbCA'BAC,$
$S \to ba \mid bBa \mid aaCa \mid aBaCa,$
$S \to baA \mid bBaA \mid aaCaA \mid aBaCaA,$
$S \to bAC \mid bBAC \mid aaCAC \mid aBaCAC,$
$A \to a \mid aA' \mid bC \mid bCA',$
$A \to bbC \mid bBbC \mid aaCbC \mid aBaCbC,$
$A \to bbCA' \mid bBbCA' \mid aaCbCA' \mid aBaCbCA',$
$A \to bAbC \mid bBAbC \mid aaCAbC \mid aBaCAbC,$
$A \to bAbCA' \mid bBAbCA' \mid aaCAbCA' \mid aBaCAbCA',$
$A' \to a \mid aA' \mid bC \mid bCA',$

$B \to b \mid bB \mid aaC \mid aBaC,$
$C \to a \mid b$

By introducing T_a and T_b, we obtain the equivalent grammar in Greibach normal form, as follows:

$S \to a \mid b \mid aA,$
$S \to aC \mid aA'C \mid bCC \mid bCA'C,$
$S \to bbCC \mid bBT_bCC \mid aT_aCT_bCC \mid aBT_aCT_bCC,$
$S \to bT_bCA'C \mid bBbT_CA'C \mid aT_aCT_bCA'C \mid aBT_aCT_bCA'C,$
$S \to bAT_bCC \mid bBAT_bCC \mid aT_aCAT_bCC \mid aBT_aCAT_bCC,$
$S \to bAT_bCA'C \mid bBAT_bCA'C \mid aT_aCAT_bCA'C \mid aBT_aCAT_bCA'C,$
$S \to aAC \mid aA'AC \mid bCAC \mid bCA'AC,$
$S \to bT_bCAC \mid bBT_bCAC \mid aT_aCbCAC \mid aBT_aCT_bCAC,$
$S \to bT_bCA'AC \mid bBT_bCA'AC \mid aT_aCT_bCA'AC \mid aBT_aCT_bCA'AC,$
$S \to bAT_bCAC \mid bBAT_bCAC \mid aT_aCAbCAC \mid aBT_aCAT_bCAC,$
$S \to bAT_bCA'AC \mid bBAT_bCA'AC \mid aT_aCAT_bCA'AC \mid aBT_aCAT_bCA'AC,$
$S \to aBC \mid aA'BC \mid bCBC \mid bCA'BC,$
$S \to bT_bCBC \mid bBT_bCBC \mid aT_aCT_bCBC \mid aBT_aCT_bCBC,$
$S \to bT_bCA'BC \mid bBT_bCA'BC \mid aT_aCT_bCA'BC \mid aBT_aCT_bCA'BC,$
$S \to bAT_bCBC \mid bBAT_bCBC \mid aT_aCAT_bCBC \mid aBT_aCAT_bCBC,$
$S \to bAbCA'BC \mid bBAT_bCA'BC \mid aT_aCAT_bCA'BC \mid aBT_aCAT_bCA'BC,$
$S \to aBAC \mid aA'BAC \mid bCBAC \mid bCA'BAC,$
$S \to bT_bCBAC \mid bBT_bCBAC \mid aT_aCT_bCBAC \mid aBT_aCT_bCBAC,$
$S \to bT_bCA'BAC \mid bBT_bCA'BAC \mid aT_aCT_bCA'BAC \mid aBT_aCT_bCA'BAC,$
$S \to bAT_bCBAC \mid bBAT_bCBAC \mid aT_aCAT_bCBAC \mid aBT_aCAT_bCBAC,$
$S \to bAT_bCA'BAC \mid bBAT_bCA'BAC \mid aT_aCAT_bCA'BAC \mid aBT_aCAT_bCA'BAC,$
$S \to bT_a \mid bBT_a \mid aT_aCT_a \mid aBT_aCT_a,$
$S \to bT_aA \mid bBT_aA \mid aT_aCT_aA \mid aBT_aCT_aA,$
$S \to bAC \mid bBAC \mid aT_aCAC \mid aBT_aCAC,$
$A \to a \mid aA' \mid bC \mid bCA',$
$A \to bT_bC \mid bBT_bC \mid aT_aCT_bC \mid aBT_aCT_bC,$
$A \to bT_bCA' \mid bBT_bCA' \mid aT_aCT_bCA' \mid aBT_aCT_bCA',$
$A \to bAT_bC \mid bBAT_bC \mid aT_aCAT_bC \mid aBT_aCAT_bC,$
$A \to bAT_bCA' \mid bBAT_bCA' \mid aT_aCAT_bCA' \mid aBT_aCAT_bCA',$
$A' \to a \mid aA' \mid bC \mid bCA',$
$B \to b \mid bB \mid aT_aC \mid aBT_aC,$
$C \to a \mid b,$
$T_a \to a, T_b \to b.$