PTC (Fall 2018) – Assignment 2

徐翔哲

2018 年 11 月 22 日

Problem 1

Give context free grammars that generate the following languages, and give a brief description of the functionality of each variable in your grammars (in natural language).

a. $\{w \in \{a,b\}^* \mid$ the number of a's in w is more than the number of b's in $w\}$

b. $\{a^i b^{2j} c^j d^k \mid i,j,k \geq 1, k \geq 2i\}$

c. $\{w \in \{a,b\}^* \mid abb$ and aab are substrings of $w\}$

d. $\{a^i b^j c^k \mid i,j,k \geq 0 \wedge i+j > k\}$

a

It can be proved that for all string w having n more a than b, it can be divided into n substrings, say $w_0 w_1 w_2 \ldots$. For each w_i the number of a is exactly one more the number of b.

$$\begin{aligned}
\boldsymbol{S} &\rightarrow \boldsymbol{AS|A} \\
\boldsymbol{A} &\rightarrow \boldsymbol{a|aB|bAA} \\
\boldsymbol{B} &\rightarrow \boldsymbol{aC|bA|\epsilon} \\
\boldsymbol{C} &\rightarrow \boldsymbol{bB|aCC|b}
\end{aligned}$$

A is the language where $\#a = \#b + 1$, B is the language where $\#a = \#b$, C is the language where $\#b = \#a + 1$.

b

$$\begin{aligned}
\boldsymbol{S} &\rightarrow \boldsymbol{aTdd} \\
\boldsymbol{T} &\rightarrow \boldsymbol{aTdd|Td|B} \\
\boldsymbol{B} &\rightarrow \boldsymbol{bbc|bbBc}
\end{aligned}$$

The functionality of the variables are clear.

c

$$\begin{aligned}
\boldsymbol{S} &\rightarrow \boldsymbol{aS|bS|Sa|Sb|T|aabb} \\
\boldsymbol{T} &\rightarrow \boldsymbol{UV|VU} \\
\boldsymbol{U} &\rightarrow \boldsymbol{aU|bU|Ua|Ub|aab} \\
\boldsymbol{V} &\rightarrow \boldsymbol{aV|bV|Va|Vb|abb}
\end{aligned}$$

U means the language whose substring contains aab, V means the language whose substring contains abb. And T means the language having aab and abb as substrings separately. Finally, consider one special case S, where the string contains $aabb$, which also has both aab and abb as substrings.

d

$$\boldsymbol{S \to aA|bB}$$
$$\boldsymbol{A \to aAc|aA|B}$$
$$\boldsymbol{B \to bB|bBc|\epsilon}$$

The main idea is that for every c in this language, there exists one related a or b. Also, since the number of a or b is strictly larger than the number of c, we can infer that for all the string in this language, there's at least one a or one b before any c is generated.

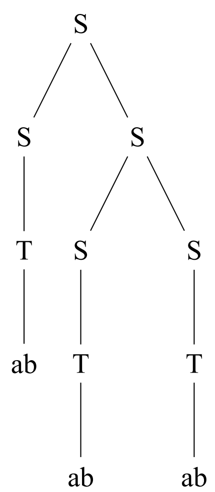
Problem 2

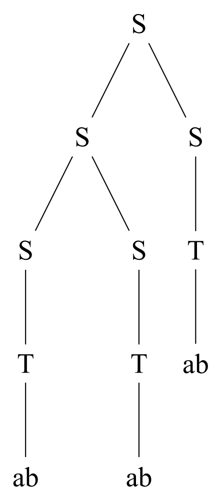
Consider the following CFG G:

$$S \to SS \mid T$$
$$T \to aTb \mid ab$$

Describe $L(G)$ and show that G is ambiguous. Give an unambiguous grammar H where $L(H) = L(G)$ and sketch a proof that H is unambiguous.

$L(G) = \{a^i b^i | i \geq 1\}^+$ Consider the string $ababab$. There exists two different derivation trees.





Define grammar H as follows:

$$\boldsymbol{S \to ST|T}$$
$$\boldsymbol{T \to aTb|ab}$$

Use structural induction:

- $\boldsymbol{T \to aTb|ab}$ is obvious unambiguous
- $\boldsymbol{S \to T}$ is obvious unambiguous
- $\boldsymbol{S \to ST}$:The variable S can only be generated on the left side of T, so for a serial of T, the only way to derivate it will be $\boldsymbol{S \to ST \to (ST)T}$. So this constructor is unambiguous.

Up to now, we have proved that every production of H is unambiguous, so H is unambiguous.

Problem 3

Let $G = (V, T, P, S)$ be a context free grammar such that $V = \{A, B\}$, $T = \{a, b\}$, $S = B$ and P is

$$B \to aAA$$
$$A \to aB \mid bB \mid a$$

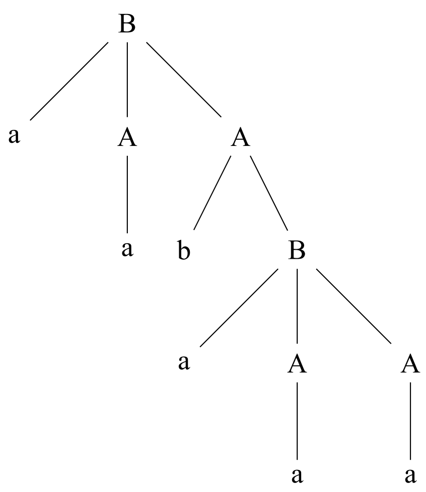
a. Give parse trees, leftmost and rightmost derivations for the following strings.

   1. $aabaaa$
   2. $abaaabaaa$

b. Convert G to a PDA that accepts the same language by empty stack.

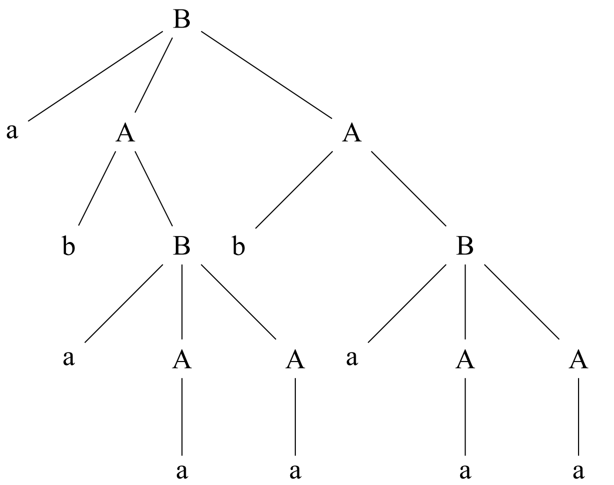
a

1



rightmost: $B \Rightarrow aAA \Rightarrow aAbB \Rightarrow aAbaAA \Rightarrow aAbaAa \Rightarrow aAbaaa \Rightarrow aabaaa$
leftmost: $B \Rightarrow aAA \Rightarrow aaA \Rightarrow aabB \Rightarrow aabaAA \Rightarrow aabaaA \Rightarrow aabaaa$

2



rightmost: $B \Rightarrow aAA \Rightarrow aAbB \Rightarrow aAbaAA \Rightarrow aAbaAa \Rightarrow aAbaaa \Rightarrow abBbaaa \Rightarrow abaAAbaaa \Rightarrow abaAabaaa \Rightarrow abaaabaaa$

leftmost: $B \Rightarrow aAA \Rightarrow abBA \Rightarrow abaAAA \Rightarrow abaaAA \Rightarrow abaaaA \Rightarrow abaaabB \Rightarrow abaaabaAA \Rightarrow abaaabaaA \Rightarrow abaaabaaa$

b

$$PDA\ P = (Q, q_0, \Sigma, \Gamma, Z_0, \delta)$$

$$Q = \{q_0\}$$

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

$$\Gamma = \{A, B, S\} \cup \Sigma$$

$$Z_0 = S$$

$$\delta(q_0, \epsilon, S) \to (q_0, B)$$

$$\delta(q_0, \epsilon, B) \to (q_0, aAA)$$

$$\delta(q_0, \epsilon, A) \to (q_0, aB)$$

$$\delta(q_0, \epsilon, A) \to (q_0, bB)$$

$$\delta(q_0, \epsilon, A) \to (q_0, a)$$

$$\delta(q_0, a, a) \to (q_0, \epsilon)$$

$$\delta(q_0, b, b) \to (q_0, \epsilon)$$

Problem 4

Begin with the grammar:

$$S \to aAa \mid bBb \mid \epsilon$$
$$A \to C \mid a$$
$$B \to C \mid b$$
$$C \to CDE \mid \epsilon$$
$$D \to A \mid B \mid ab$$

a. Eliminate ϵ-productions.

b. Eliminate any unit productions in the resulting grammar of (a.).

c. Eliminate any useless symbols in the resulting grammar of (b.).

d. Put the resulting grammar of (c.) into Chomsky normal form.

a

The nullable variables are C, S, A, B, D, then we can get:

$$\boldsymbol{S \to aAa|bBb|aa|bb}$$
$$\boldsymbol{A \to C|a}$$
$$\boldsymbol{B \to C|b}$$
$$\boldsymbol{C \to CDE|CE|DE|E}$$
$$\boldsymbol{D \to ab|A|B}$$

b

$$\boldsymbol{S \to aAa|bBb|aa|bb}$$
$$\boldsymbol{A \to CDE|CE|DE|a}$$
$$\boldsymbol{B \to CDE|CE|DE|b}$$
$$\boldsymbol{C \to CDE|CE|DE}$$
$$\boldsymbol{D \to ab|a|CDE|CE|DE|b}$$

c

$$\boldsymbol{S \to aAa|bBb|aa|bb}$$
$$\boldsymbol{A \to a}$$
$$\boldsymbol{B \to b}$$

d

$$\boldsymbol{S \to AT_1|BT_2|AA|BB}$$
$$\boldsymbol{T_1 \to AA}$$
$$\boldsymbol{T_2 \to BB}$$
$$\boldsymbol{A \to a}$$
$$\boldsymbol{B \to b}$$

Problem 5

Given grammar G:

$$S \to AB \mid BC$$
$$A \to BA \mid a$$
$$B \to CC \mid b$$
$$C \to AB \mid a$$

Please use CYK algorithm to decide whether string $ababa$ belongs to $L(G)$.

ababa[0:0]:{A,C}

ababa[1:1]:{B}

ababa[2:2]:{A,C}

ababa[3:3]:{B}

ababa[4:4]:{A,C}

ababa[0:1]:{S,C}

ababa[1:2]:{A,S}

ababa[2:3]:{S,C}

ababa[3:4]:{A,S}

ababa[0:2]:{B}

ababa[1:3]:{S,C}

ababa[2:4]:{B}

ababa[0:3]:{B}

ababa[1:4]:{B}
ababa[0:4]:{S,A}
Since $ababa[0:4]$ contains S, this string belongs to $L(G)$

Problem 6

Use the CFL pumping lemma to show each of these languages are not context free.

a. $\{a^i b^j c^k \mid i < j < k\}$

b. $\{0^p \mid p \text{ is a prime}\}$

c. $\{ww^R w \mid w \in \{0,1\}^*\}$

a

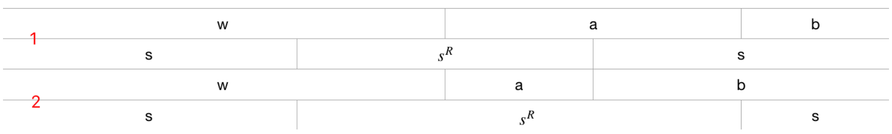
$\forall n$, choose $a^n b^{n+1} c^{n+2}$. Since $|vwx| \leq n$, so $|vx|$ contains at least one of character a, b and c but not all.
If vx doesn't contain c, we repeat vx for $2n$ times. In such case, the number of a or b is larger than the number of c.
If vx contains c, then it cannot contain a. Then we remove vx from the origin string, which will cause either the number of b is not larger than the number of a or the number of c is not larger than the number of b.

b

Suppose that $|vx| = q$, then we repeat $|vx|$ for $p+1$ times. Then $|uv^{p+1}wx^{p+1}y| = p + p \cdot q = (1+q)p$, which is not a prime. So $uv^{p+1}wx^{p+1}y$ is not in this language.

c

Let $|w| > n$, also, choose w such that every w's prefix is not w's suffix. Since $|vwx| < n$, so $|vx|$ can't contain both the beginning and the ending of w. Suppose that uwy=wab, where a isi one prefix of w^R and b is one suffix of w. If wab can be written as ss^Rs, it should be one of the following case:



case1 means suffix(s^R)=prefix(a) $\Rightarrow$ suffix(w^R)=prefix(w^R) $\Rightarrow$ prefix(w)=suffix(w), which is a contradiction with our hypothesis.
case2 means w's suffix being w's prefix, which is a contradiction with our hypothesis.

Problem 7

Show that the CFL is closed under the following operations:

a. $init(L) = \{w \mid \text{for some } x, wx \in L\}$. (**Hint**: Start with a CNF grammar for the language L)

b. $cycle(L) = \{xy \mid yx \in L\}$ (**Hint**: Try a PDA-based construction)

a

For all the production:
If $\boldsymbol{P} \rightarrow \boldsymbol{\alpha}$, convert it to $\boldsymbol{P'} \rightarrow \boldsymbol{\alpha}|\boldsymbol{\epsilon}$;
if $\boldsymbol{P} \rightarrow \boldsymbol{AB}$, convert it to $\boldsymbol{P'} \rightarrow \boldsymbol{AB'}|\boldsymbol{A'}$
Proof: Induction on the steps of derivation.
Basis: $\boldsymbol{P} \rightarrow \boldsymbol{\alpha}$, the P' is all the prefix of $\alpha(\alpha$ and $\epsilon)$.
Induction: For all derivation within n step, the new grammar will generate all the prefix of the strings that can be derived within n steps.
For the derivation of step n+1, we first make one step.
If the first step is $\boldsymbol{P'} \rightarrow \boldsymbol{AB'}$, then B' contains all the prefix of B since it can be derived within n steps. Then we can get that P' contains all the prefix of P.
If the first step is $\boldsymbol{P'} \rightarrow \boldsymbol{A'}$, then A' contains all the prefix of A since it can be derived within n steps. Then we can get that P' contains all the prefix of P.
Now we have proved that all the strings the new grammar generate are the prefix of strings in the origin language.

b

不会做……给助教学长拜个早年

Problem 8

If L is regular, it satisfies pumping lemma for sure! But if a language satisfies pumping lemma, is it regular? Prove or disprove it.
$(abc)^{+}0^{n}1^{n}$, choose n=4, and let $x = \epsilon,\ y = abc$. However, this language is obviously not a regular language.