

PTC (Fall 2018) – Assignment 2

徐翔哲

2018 年 11 月 21 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).

- $\{w \in \{a, b\}^* \mid \text{the number of } a\text{'s in } w \text{ is more than the number of } b\text{'s in } w\}$
- $\{a^i b^{2j} c^j d^k \mid i, j, k \geq 1, k \geq 2i\}$
- $\{w \in \{a, b\}^* \mid abb \text{ and } aab \text{ are substrings of } w\}$
- $\{a^i b^j c^k \mid i, j, k \geq 0 \wedge i + j > k\}$

a

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

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

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

b

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

The functionality of the variables are clear.

c

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

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

$$S \rightarrow aA|bB$$

$$A \rightarrow aAc|aA|B$$

$$B \rightarrow bB|bBc|\epsilon$$

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

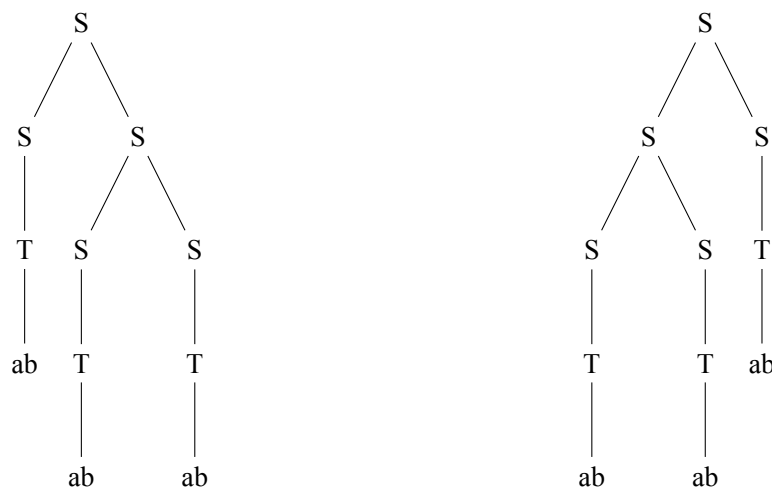
Consider the following CFG G :

$$S \rightarrow SS | T$$

$$T \rightarrow aTb | ab$$

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:

$$S \rightarrow ST|T$$

$$T \rightarrow aTb|ab$$

Use structural induction:

- $T \rightarrow aTb|ab$ is obvious unambiguous
- $S \rightarrow T$ is obvious unambiguous
- $S \rightarrow ST$: 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 $S \rightarrow ST \rightarrow (ST)T$. 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

$$B \rightarrow aAA$$

$$A \rightarrow aB \mid bB \mid a$$

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

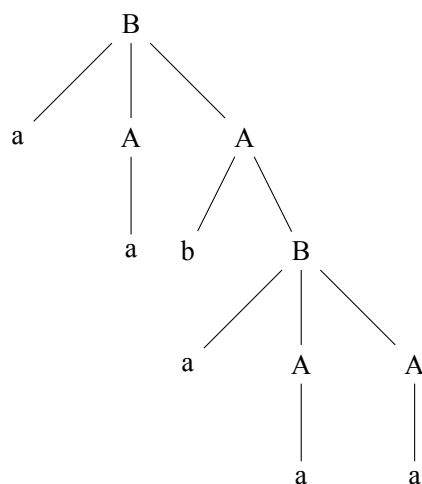
1. *aabaaa*

2. *abaaabaaa*

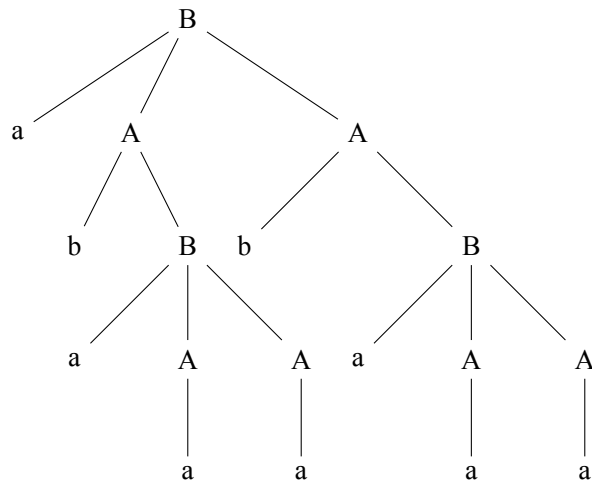
b. Convert G to a PDA that accepts the same language by empty stack.

a

1



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


$$\begin{aligned} \text{rightmost: } B &\Rightarrow aAA \Rightarrow aAbB \Rightarrow aAbaAA \Rightarrow aAbaAa \Rightarrow aAbaaa \Rightarrow abBbaaa \Rightarrow \\ &abaAbaaaa \Rightarrow abaAabaaa \Rightarrow abaaabaaa \end{aligned}$$

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

b

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

$$Q = \{q_0\}$$

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

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

$$Z_0 = S$$

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

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

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

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

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

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

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

Problem 4

Begin with the grammar:

$$S \rightarrow aAa \mid bBb \mid \epsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow C \mid b$$

$$C \rightarrow CDE \mid \epsilon$$

$$D \rightarrow A \mid B \mid ab$$

- 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:

$$S \rightarrow aAa \mid bBb \mid aa \mid bb$$

$$A \rightarrow C \mid a$$

$$B \rightarrow C \mid b$$

$$C \rightarrow CDE \mid CE \mid DE \mid E$$

$$D \rightarrow ab \mid A \mid B$$

b

$$S \rightarrow aAa \mid bBb \mid aa \mid bb$$

$$A \rightarrow CDE \mid CE \mid DE \mid E \mid a$$

$$B \rightarrow CDE \mid CE \mid DE \mid E \mid b$$

$$C \rightarrow CDE \mid CE \mid DE \mid E$$

$$D \rightarrow ab \mid CDE \mid CE \mid DE \mid E \mid a \mid CDE \mid CE \mid DE \mid E \mid b$$

c

$$S \rightarrow aAa|bBb|aa|bb$$

$$A \rightarrow a$$

$$B \rightarrow b$$

d

$$S \rightarrow AT_1|BT_2|AA|BB$$

$$T_1 \rightarrow AA$$

$$T_2 \rightarrow BB$$

$$A \rightarrow a$$

$$B \rightarrow b$$

Problem 5

Given grammar G :

$$S \rightarrow AB \mid BC$$

$$A \rightarrow BA \mid a$$

$$B \rightarrow CC \mid b$$

$$C \rightarrow AB \mid a$$

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}

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.

- $\{a^i b^j c^k \mid i < j < k\}$
- $\{0^p \mid p \text{ is a prime}\}$
- $\{ww^R w \mid w \in \{0, 1\}^*\}$

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 $uwv = wab$, where a is one prefix of w^R and b is one suffix of w . If wab can be written as $ss^R s$, it should be one of the following case:

	w	a	b
1	s	s^R	s
2	w	a	b
	s	s^R	s

case1 means $\text{suffix}(s^R) = \text{prefix}(a) \Rightarrow \text{suffix}(w^R) = \text{prefix}(w^R) \Rightarrow \text{prefix}(w) = \text{suffix}(w)$, 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 $P \rightarrow \alpha$, convert it to $P' \rightarrow \alpha|\epsilon$;

if $P \rightarrow AB$, convert it to $P' \rightarrow AB'|A'$

Proof: Induction on the steps of derivation.

Basis: $P \rightarrow \alpha$, the P' is all the prefix of α (α and ϵ).

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 $P' \rightarrow AB'$, 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 $P' \rightarrow A'$, 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

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.

$(abc)^+0^n1^n$, choose $n=4$, and let $x = \epsilon$, $y = abc$. However, this language is obviously not a regular language.