Recitation 6 - Context-Free Languages and Homework 3

John Chilton

June 8, 2007

- Homework 3
- Converting a CFG to Chomsky Normal Form
- Converting a CFG to a PDA

Problem 1. Provide CFGs for two languages.

- Just need show a CFG

Problem 1. Provide CFGs for two languages.

- Just need show a CFG
- Don't need to prove it works or provide a formal definition

Problem 1. Provide CFGs for two languages.

- Just need show a CFG
- Don't need to prove it works or provide a formal definition
- Use the notation that upper case letters are non-terminals and lower case letters are terminals.

Problem 1. Provide CFGs for two languages.

- Just need show a CFG
- Don't need to prove it works or provide a formal definition
- Use the notation that upper case letters are non-terminals and lower case letters are terminals.
- These may be challenging, start with simpler grammars

$$L_1 = \{a^m b^n c^p d^q | m + n = p + q\}$$

$L_1 = \{a^m b^n c^p d^q | m + n = p + q\}$

- $aabbcddd \in L_1$, $aaaccd \in L_1$, $ad \in L_1$ $\epsilon \in L_1$

$L_1 = \{a^m b^n c^p d^q | m + n = p + q\}$

- $aabbcddd \in L_1$, $aaaccd \in L_1$, $ad \in L_1$ $\epsilon \in L_1$
- $abcdd \notin L_1$, $acbd \notin L_1$

$L_1 = \{a^m b^n c^p d^q | m + n = p + q\}$

- $aabbcddd \in L_1$, $aaaccd \in L_1$, $ad \in L_1$ $\epsilon \in L_1$
- $abcdd \notin L_1$, $acbd \notin L_1$
- **Do not** use "rules" like $S \rightarrow a^m S$, number of terminals appearing in a rule must be constant

$L_1 = \{a^m b^n c^p d^q | m + n = p + q\}$

- $aabbcddd \in L_1$, $aaaccd \in L_1$, $ad \in L_1$ $\epsilon \in L_1$
- $abcdd \notin L_1$, $acbd \notin L_1$
- **Do not** use "rules" like $S \rightarrow a^m S$, number of terminals appearing in a rule must be constant
- Build up to L_1, first try to come up with CFGs that generate $a^n c^n$ and then $\{a^m c^p d^q \mid m = p + q\}$.

$L_2 = \{x_1\#x_2\#\ldots\#x_k \mid x_i \in \{a,b\}^*$ and for some i and j, $x_i = x_j^R\}$

- $aab\#bbb\#baa\#bab \in L_2$

$L_2 = \{x_1\#x_2\#\ldots\#x_k \mid x_i \in \{a,b\}^*$ and for some i and j, $x_i = x_j^R\}$

- $aab\#bbb\#baa\#bab \in L_2$
- $baa\#baba\#ababa \notin L_2$

$L_2 = \{x_1\#x_2\#\ldots\#x_k \mid x_i \in \{a,b\}^*$ and for some i and j, $x_i = x_j^R\}$

- $aab\#bbb\#baa\#bab \in L_2$
- $baa\#baba\#ababa \notin L_2$
- Is $aba\#baa$ a member of L_2

$L_2 = \{x_1\#x_2\#\ldots\#x_k \mid x_i \in \{a,b\}^*$ and for some i and j, $x_i = x_j^R\}$

- $aab\#bbb\#baa\#bab \in L_2$
- $baa\#baba\#ababa \notin L_2$
- Is $aba\#baa$ a member of L_2
- Build up to solving L_2, start with $x\#x^R$, and then 2.6 part c.

Problem 2. Prove

$$S \rightarrow aSb \mid bY \; Ya$$
$$Y \rightarrow bY \mid aY \mid \epsilon$$

generates the language. $\{w \mid w \text{ is not of the form } a^n b^n \text{ for some } n\}$

$$S \rightarrow aSb \mid bY \;\; Ya$$
$$Y \rightarrow bY \mid aY \mid \epsilon$$

- $L := \{w \mid w \text{ is not of the form } a^n b^n \text{ for some } n\}$
- Let L_S be all the strings that can be derived from S, and L_Y be all the strings that can derived from Y.
- Show $L_S \subseteq L$ and $L \subseteq L_S$.
- Use induction on number of steps in derivation to show $L_S \subseteq L$.
- Use induction on length of string to show $L \subseteq L_S$.

Problem 3. Convert CFGs from problem 1 into PDAs. Just follow the algorithm laid out on pages 115-118. We will go through this today. Even if you can't get something for problem 1, put something reasonable down and then do this step with that. This is one of the easier problems.

Problem 4. Given a CFG convert it into Chompsky Normal form. Just use the algorithm laid out in Theorem 2.9 on page 107. This algorithm is demonstrated on page 108.

Problem 5. Show that the class of context-free languages is closed under union, concatenation, and start. Experiment with *very simple* grammars to figure out the pattern.

▷ Let L_1 and L_2 be two given context-free languages. Because these languages are context-free there exists context-free grammars which generate these languages. Let $G_1 = (V_1, \Sigma, R_1, S_1)$ generate L_1, and let $G_2 = (V_2, \Sigma, R_2, S_2)$ generate L_2. Now consider with following grammar which generates $L_1 \cup L_2$ ($L_1 \circ L_2$),
$G = (V, \Sigma, R, S)$, with:
(describe V, R, and S)

$\triangleright$ Let L a given context-free languages. Because this languages is context-free there exists a context-free grammar which generates L. Let $G = (V, \Sigma, R, S)$ generate L. Now consider with following grammar which generates L^*, $G' = (V', \Sigma, R', S')$, with:
(describe V', R', and S')

Problem 6. Convert the given CFG to a PDA. See notes for problem 3. Do this problem!

Problem 7. Consider $A = \{a^m b^n c^n\}$ and $B = \{a^n b^n c^m\}$, use these along with the fact that Example 2.36 ($\{a^n b^n c^n\}$) is not context-free to show that context-free languages are not closed under intersection.
(Start with this problem.)

Problem 8. Let L be a given context-free language and R be a given regular language.

- Part 1. Show $L - R$ must be context-free. (Use the result that $C \cap S$, is context-free for any context-free C and regular S.)
- Part 2. Show $R - L$ isn't necessarily context-free. (Use the result that context-free grammars are not closed under complementation.)
- Hint: In set notation form, recall the definition of $A - B$.

Problem 9. Use the pumping lemma for context-free languages to show three languages are not context-free. I will demonstrate examples of using the pumping lemma for context-free grammars next week.

Problem 10. Let G be a CFG in Chomsky normal form that contains b variables. Show that, if G generates some string with a derivation having at least 2^b steps, $L(G)$ contains an infinite number of strings.

Construct a grammar that generates the following language:

$$\{u\#v \mid u, v \in \{a, b\}^* \text{ and are even length palindromes}\}$$

In Chomsky Normal Form, every rule is of the form: $A \longrightarrow BC$ or $A \longrightarrow d$. Where B and C are variables that are not the start symbol, and d is a terminal. Additionally the rule $S \longrightarrow \epsilon$ is allowed if S is the start symbol.

- Replace the start symbol
- Replace $S \longrightarrow \epsilon$ rules
- Replace unit rules of the form $A \longrightarrow B$
- Break up all rules of the form $A \longrightarrow u_1 u_2 \ldots u_k$ for $k \geq 3$.
- Replace all terminals in rules of the form $A \longrightarrow u_1 u_2$ with variable

Consider the CFG:

$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

Replace start symbol.

$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

Replace start symbol.

$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

$$\downarrow$$

$$S_0 \rightarrow S$$
$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

Replace ϵ transitions.

$$S_0 \rightarrow S$$
$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

Replace ϵ transitions.

$$\begin{aligned} S_0 &\to S \\ S &\to T\#T \\ T &\to aTa \mid bTb \mid \epsilon \end{aligned}$$

$$\downarrow$$

$$\begin{aligned} S_0 &\to S \\ S &\to T\#T \mid \#T \mid T\# \mid \# \\ T &\to aTa \mid bTb \mid aa \mid bb \end{aligned}$$

Replace unit rules.

$$S_0 \to S$$
$$S \to T\#T \mid \#T \mid T\# \mid \#$$
$$T \to aTa \mid bTb \mid aa \mid bb$$

Replace unit rules.

$$\begin{aligned} S_0 &\rightarrow S \\ S &\rightarrow T\#T \mid \#T \mid T\# \mid \# \\ T &\rightarrow aTa \mid bTb \mid aa \mid bb \end{aligned}$$

$$\downarrow$$

$$\begin{aligned} S_0 &\rightarrow T\#T \mid \#T \mid T\# \mid \# \\ S &\rightarrow T\#T \mid \#T \mid T\# \mid \# \\ T &\rightarrow aTa \mid bTb \mid aa \mid bb \end{aligned}$$

Cut each rule to size at most 2.

$$S_0 \rightarrow T\#T \mid \#T \mid T\# \mid \#$$
$$S \rightarrow T\#T \mid \#T \mid T\# \mid \#$$
$$T \rightarrow aTa \mid bTb \mid aa \mid bb$$

Cut each rule to size at most 2.

$$\begin{aligned} S_0 &\to T\#T \mid \#T \mid T\# \mid \# \\ S &\to T\#T \mid \#T \mid T\# \mid \# \\ T &\to aTa \mid bTb \mid aa \mid bb \end{aligned}$$

$$\downarrow$$

$$\begin{aligned} S_0 &\to TT_1 \mid \#T \mid T\# \mid \# \\ S &\to TT_1 \mid \#T \mid T\# \mid \# \\ T &\to aT_2 \mid bT_3 \mid aa \mid bb \\ T_1 &\to \#T \\ T_2 &\to Ta \\ T_3 &\to Tb \end{aligned}$$

Replace terminals.

$$S_0 \to TT_1 \mid \#T \mid T\# \mid \#$$
$$S \to TT_1 \mid \#T \mid T\# \mid \#$$
$$T \to aT_2 \mid bT_3 \mid aa \mid bb$$
$$T_1 \to \#T \quad T_2 \to Ta \quad T_3 \to Tb$$

Replace terminals.

$$
\begin{aligned}
S_0 &\to TT_1 \mid \#T \mid T\# \mid \# \\
S &\to TT_1 \mid \#T \mid T\# \mid \# \\
T &\to aT_2 \mid bT_3 \mid aa \mid bb \\
T_1 &\to \#T \quad T_2 \to Ta \quad T_3 \to Tb
\end{aligned}
$$

$$\downarrow$$

$$
\begin{aligned}
S_0 &\to TT_1 \mid PT \mid TP \mid \# \\
S &\to TT_1 \mid PT \mid TP \mid \# \\
T &\to AT_2 \mid BT_3 \mid AA \mid BB \\
T_1 &\to PT \quad T_2 \to TA \quad T_3 \to TB \\
A &\to a \quad B \to b \quad P \to \#
\end{aligned}
$$

A convoluted example to illustrate some perils:

$$A \longrightarrow aAa \mid B \mid \epsilon$$
$$B \longrightarrow bBb \mid A$$

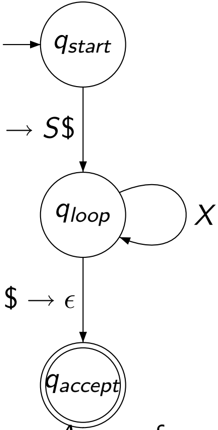
Remove ϵ rules.

A convoluted example to illustrate some perils:

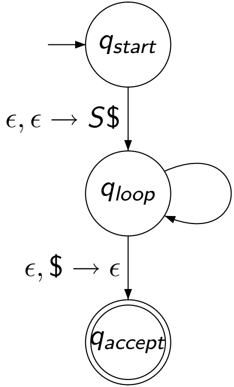
$$A \rightarrow aAa \mid B \mid \epsilon$$
$$B \rightarrow bBb \mid A$$

Remove ϵ rules.
This could be a problem when removing unit rules like $A \rightarrow B$ also!

Outline of Lemma 2.21:



X includes transitions of the form $\epsilon, A \rightarrow w$ for each rule $A \rightarrow w$, and $a, a \rightarrow \epsilon$ for each terminal a.



Try this for the grammar:

$$S \rightarrow T\#T$$
$$T \rightarrow aTa \mid bTb \mid \epsilon$$

Some other problems:

$$a^n b^m c^n$$

$$\{w\#x \mid w^R \text{ is a substring of } x\}$$

$$\{w \mid |w| \text{ is odd and middle symbol is } 0\}$$

$$\{w \mid w \text{ contains at least three 1s}\}$$

$$\{w \mid w \text{ starts and ends with same character }\}$$

$$\phi$$