Ex3! $C = \{ w \mid w \text{ contains twice as many a's as b's} \}$

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

ε	ab	aab	aba	baa
✓	✗	✓	✓	✓

a
✗

aab   aba   baa

_ b _ , _ _ b

b _ _

○ a ○ a ○ b ○

○ a ○ b ○ a ○

○ b ○ a ○ a ○

$$S \rightarrow SaSaSbS \mid SaSbSaS \mid SbSaSaS \mid \varepsilon$$

* Remember
  - All regular languages can be expressed by context free grammar
  - Essentially, CFG is more powerful than regular expressions

Ambiguity

Sometimes a grammar can generate the same string in several different ways.

Such strings will have several different parse trees.

In some applications, this may be undesirable.

If a grammar generates a string in several different ways, we say that string is derived ambiguously in that grammar.

If a grammar generates some string ambiguously, we say that the grammar is ambiguous.

Example:

$$\langle E \rangle \to \langle E \rangle + \langle E \rangle \mid \langle E \rangle \times \langle E \rangle \mid (\langle E \rangle) \mid a$$

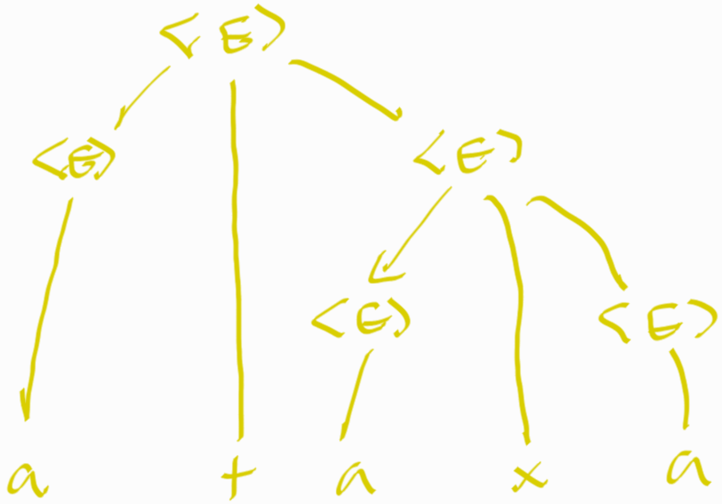
$V = \{ \langle E \rangle \}$ $\Sigma = \{a, (,), +, \times\}$

$S = \langle E \rangle$

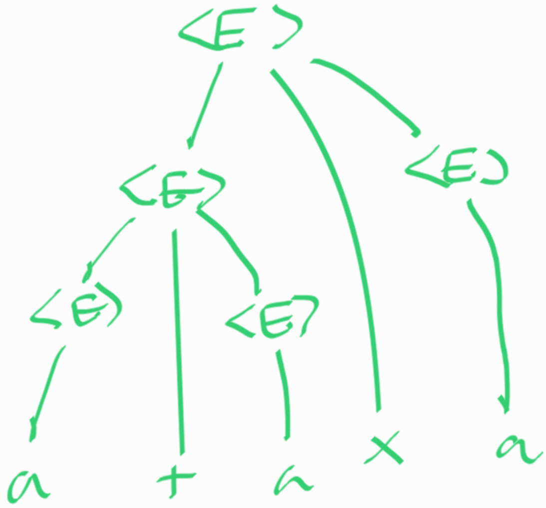
$a + a \times a$

1. $\langle E \rangle \Rightarrow \langle E \rangle + \langle E \rangle \Rightarrow \langle E \rangle + \langle E \rangle \times \langle E \rangle \overset{*}{\Rightarrow} a + a \times a$

2. $\langle E \rangle \Rightarrow \underline{\langle E \rangle} \times \langle E \rangle \Rightarrow \langle E \rangle + \langle E \rangle \times \langle E \rangle \overset{*}{\Rightarrow} a + a \times a$



<E>
<E> + <E>
<E> x <E>
a + a x a



<E>
<E> x <E>
<E> + <E>
a + a x a

Chomsky Normal Form (CNF)

When working with CFG, it is often convenient to have them in simplified form.

This useful, when we have algorithms working on CFGs.

Definition

Chomsky Normal Form (CNF)

context free grammer is in CNF if every rule is in the form of:

$$A \longrightarrow BC$$

$$A \longrightarrow a$$

a is a terminal, and A, B, C are any variable except that B & C may not be start variable.

we also allow $S \rightarrow \varepsilon$ for the start variable (only for the start variable)

Theorem 2.9

Any context free language is generated by a context-free grammar in chomsky Normal form.

Proof: (By construction)

Instead of going through detailed proof, we will go through an example to understand this.

$S \to ASA \mid aB$

$A \to B \mid S$

$B \to b \mid \varepsilon$

Let's convert this to CNF

1. We add a new starting variable.
   (This guarantees that the start variable does not appear on right hand side of any rule.

   ex: $S_0 \longrightarrow S$

2. We take care of ε-rules.
   we remove $A \to \varepsilon$ rules where A is not the start state.

   for each occurrence of A on the right hand side of a rule, we add a new rule with that occurrence deleted.

want to remove $A \to \varepsilon$

Ex: $R \longrightarrow uAvAw$

* $R \longrightarrow uAvAw \mid uvAw \mid uAvw \mid uvw$

$R \to A \Rightarrow R \to \varepsilon$ (only if we did not remove $R \to \varepsilon$)

3. Handle all unit rules

(i) we remove unit rule $A \rightarrow B$

(ii) when we see $B \rightarrow u$, we add $A \rightarrow u$ unless we have already removed it earlier

* u is a string of variables and terminals.

4. Convert all remaining rules into proper form.

(i) we replace each rule in the form of

$A \rightarrow u_1 u_2 \dots u_k$, where $k \geq 3$, and u_i is a variable or a terminal symbol, with rules

$A \rightarrow u_1 A_1$

$A_1 \rightarrow u_2 A_2$

$A_2 \rightarrow u_3 A_3$ ...... $A_{k-2} \rightarrow u_{k-1} u_k$

we replace any u_i in the preceding rule(s) with the variable U_i and

$U_i \rightarrow u_i$

$$A \rightarrow u_1 u_2 u_3 \cdots u_k$$

$$A \rightarrow u_1 A_1$$
$$A_1 \rightarrow u_2 u_3 \cdots u_k$$

$$A \rightarrow u_1 A_1$$
$$A_1 \rightarrow u_2 A_2$$
$$A_2 \rightarrow u_3 u_4 \cdots u_k$$

$$\vdots$$

Let's do an example

$S \longrightarrow ASA \mid aB$

$A \longrightarrow B \mid S$

$B \longrightarrow b \mid \varepsilon$

$S_0 \longrightarrow S$

$S \longrightarrow ASA \mid a\underline{B}$

$A \longrightarrow \underline{B} \mid S$

$B \longrightarrow b \mid \cancel{\varepsilon}$

Remove $B \longrightarrow \varepsilon$

$S_0 \longrightarrow S$

$S \longrightarrow ASA \mid aB \mid a$

$A \longrightarrow B \mid \cancel{\varepsilon} \mid S$

$B \longrightarrow b$

---

remove $A \longrightarrow \varepsilon$

$S_0 \longrightarrow S$

$S \longrightarrow ASA \mid SA \mid AS \mid S \mid aB \mid a$

$A \longrightarrow \cancel{B} \mid S$

$B \longrightarrow b$

remove $A \to B$

$S_0 \to S$

$S \to ASA \mid SA \mid AS \mid S \mid aB \mid a$

$A \to \cancel{S} \mid b$

$B \to b$

---

remove $A \to S$

$S_0 \to S$

$S \to ASA \mid SA \mid AS \mid \cancel{S} \mid aB \mid a$

$A \to ASA \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

---

remove $S \to S$

$S_0 \to \cancel{S}$

$S \to ASA \mid SA \mid AS \mid aB \mid a$

$A \to ASA \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

---

remove $S_0 \to S$

$\underline{S_0} \to \underline{ASA} \mid SA \mid AS \mid aB \mid a$

$S \to ASA \mid SA \mid AS \mid aB \mid a$

$A \to ASA \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

$S_0 \to AU_1 \mid SA \mid AS \mid aB \mid a$

$U_1 \to SA$

$S \to ASA \mid SA \mid AS \mid aB \mid a$

$A \to ASA \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

fixing improper rules.

$S_0 \to AU_1 \mid SA \mid AS \mid aB \mid a$

$U_1 \to SA$

$S \to AU_2 \mid SA \mid AS \mid aB \mid a$

$U_2 \to SA$

$A \to ASA \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

$S_0 \to AU_1 \mid SA \mid AS \mid aB \mid a$

$U_1 \to SA$

$S \to AU_2 \mid SA \mid AS \mid aB \mid a$

$U_2 \to SA$

$A \to AU_1 \mid SA \mid AS \mid aB \mid a \mid b$

$B \to b$

$S_0 \rightarrow AU_1 \mid SA \mid AS \mid XB \mid a$

$U_1 \rightarrow SA$

$S \rightarrow AU_2 \mid SA \mid AS \mid XB \mid a$

$U_2 \rightarrow SA$

$A \rightarrow AU_1 \mid SA \mid AS \mid XB \mid a \mid b$

$B \rightarrow b$

$X \rightarrow a$

All the rules follow the CNF form.