Feb 17

Q: What is the difference between the CFG's that Hunter taught us in 305 and the ones covered here in 338?

A: Good question!

- In a nutshell, the CFG's we are covering in 338 are more general and focus more on logic than format.
- On the other hand, the CFG's covered in 305 are deterministic, while CFG's covered here are nondeterministic. Hence the CFG's covered in 305 is only a subset of what we are covering — you'll see more details next week.

Ambiguity

If a grammar generates the same string in several different ways, we say the grammar (string) is ambiguous.

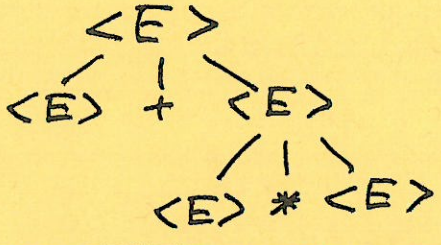
Ex. $\langle E \rangle \to \langle E \rangle + \langle E \rangle \mid \langle E \rangle * \langle E \rangle \mid (\langle E \rangle) \mid a \mid b$

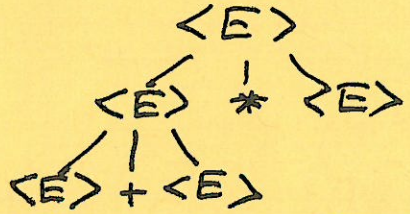
$\langle E \rangle \Rightarrow \langle E \rangle + \underline{\langle E \rangle} \Rightarrow \langle E \rangle + \langle E \rangle * \langle E \rangle$

$\overset{*}{\Rightarrow}\ a + b * b$

$\langle E \rangle \Rightarrow \underline{\langle E \rangle} * \langle E \rangle \Rightarrow \langle E \rangle + \langle E \rangle * \langle E \rangle$

$\overset{*}{\Rightarrow}\ a + b * b$





Chomsky Normal Form (CNF)

A context-free grammar is in CNF if every rule is in the form of

$A \to BC$

$A \to a$

regular:

$S \to a$

$S \to aT$

Thm 2.9

Any context-free language is generated by a context-free grammar in CNF.

Proof: (By Construction)

We'll follow an example.

$$\begin{cases} S \to ASA \mid aB \\ A \to B \mid S \\ B \to b \mid \varepsilon \end{cases}$$

① Add $S_0 \to S$, where S is the original start variable.

$$\begin{cases} S_0 \to S \\ S \to ASA \mid aB \\ A \to B \mid S \\ B \to b \mid \varepsilon \end{cases}$$

② Remove ε-rules, i.e., $A \to \varepsilon$. We need to handle every occurrence of A.

Ex. $\begin{cases} R \to uAvAw \\ A \to \varepsilon \end{cases}$

We have 3 possibilities: $R \to uvAw$, $R \to uAvw$, $R \to uvw$.

$\times B \to \varepsilon$

$$\begin{cases} S_0 \to S \\ S \to ASA \mid aB \mid a \\ A \to B \mid S \mid \varepsilon \\ B \to b \end{cases}$$

$\times A \to \varepsilon$

$$\begin{cases} S_0 \to S \\ S \to ASA \mid aB \mid a \mid SA \mid AS \mid S \\ A \to B \mid S \\ B \to b \end{cases}$$

(3) Remove unit rules $A \to B$

× $S \to S$

$$\begin{cases} S_0 \to S \\ S \to ASA \mid aB \mid a \mid SA \mid AS \\ A \to B \mid S \\ B \to b \end{cases}$$

× $S_0 \to S$

$$\begin{cases} S_0 \to ASA \mid aB \mid a \mid SA \mid AS \\ S \to ASA \mid aB \mid a \mid SA \mid AS \\ A \to B \mid S \\ B \to b \end{cases}$$

× $A \to B$

$$\begin{cases} S_0 \to ASA \mid aB \mid a \mid SA \mid AS \\ S \to ASA \mid aB \mid a \mid SA \mid AS \\ A \to S \mid b \\ B \to b \end{cases}$$

× $A \to S$

$$\begin{cases} S_0 \to ASA \mid aB \mid a \mid SA \mid AS \\ S \to ASA \mid aB \mid a \mid SA \mid AS \\ A \to b \mid ASA \mid aB \mid a \mid SA \mid AS \\ B \to b \end{cases}$$

(4) Convert the remaining rules into CNF by adding variables: $A \to U_1 U_2 U_3 U_4$

$\Rightarrow A \to U_1 A_1,\ A_1 \to U_2 A_2,\ A_2 \to U_3 U_4$

(A_1, A_2, A are new variables!)

$$\begin{cases} S_0 \to AA_1 \mid aB \mid a \mid SA \mid AS \\ S \to AA_1 \mid aB \mid a \mid SA \mid AS \\ A \to b \mid AA_1 \mid aB \mid a \mid SA \mid AS \\ A_1 \to SA \\ B \to b \end{cases}$$ // A_1 is new

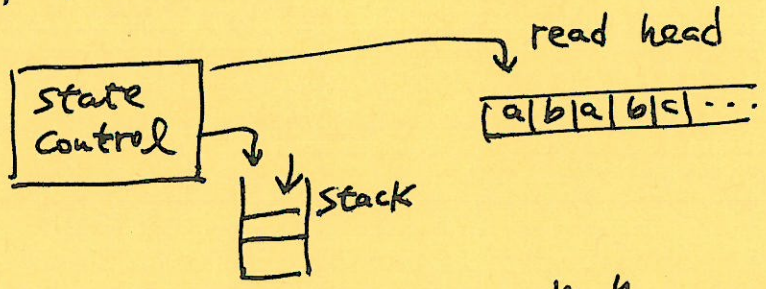
⇩

$$\begin{cases} S_0 \to AA_1 \mid UB \mid a \mid SA \mid AS \\ S \to AA_1 \mid UB \mid a \mid SA \mid AS \\ A \to b \mid AA_1 \mid UB \mid a \mid SA \mid AS \\ A_1 \to SA \\ B \to b \\ U \to a \end{cases}$$ // U is new

□

2.2 Push down automata (PDA or pda).

NFA + stack.



Q: How do we recognize 0^n1^n with a PDA?