Lecture 6: Context Free Languages

- Regular languages can be described by regexes, but some languages are not regular - \( L_1 \) is not regular

\[ L_1 = \{ 0^n 1^n \} \]

- Rules to generate \( L_1 \):
  1. \( S \rightarrow \varepsilon \) or \( S \rightarrow 0S1 \)

Generate \( 000111 \) using these rules: \( S \rightarrow 0S1 \rightarrow 00S11 \rightarrow 000111 \)

\[ L_2 = \{ w \mid w \text{ is a regex over } \{0,1,\varepsilon\} \text{ w consists of } 0, 1, \varepsilon, (, ), \cup, \ast, * \} \]

- \( L_2 \) not regular language because there must be same # of left & right parenthesis in \( w \) to be valid regex.

- Context free grammar for \( L_2 \):

\[
S \rightarrow e | 0 | 1 | \varepsilon | U | C | K
\]

\[
U \rightarrow (S) U (S)
\]

\[
C \rightarrow (S) \ast (S) | A
\]

\[
A \rightarrow 0 | 1 | OA | (A)
\]

\[
K \rightarrow (S) \ast | 0 \ast | 1 \ast
\]

Generate \( (010^*) U ((1^* 0^*) \cup \varepsilon) \)

\[ S \Rightarrow U \Rightarrow (S) U (S) \Rightarrow (C) U (S) \Rightarrow ((S) \ast (S)) U (S) \]

(}\)
- definition for context free grammar: \( (CFG) \)

A \( CFG \) is a 4-tuple: \( (V, \Sigma, R, S) \)
where \( V \) is a finite set of variables
\( \Sigma \) is a finite set of terminals (the alphabet)
\( R \) is a set of rules of the form \( A \rightarrow U \) where
\( A \in V, U \in (V \cup \Sigma)^* \)
\( S \in V \) is the start variable

If \( u, v, w \) are strings over \( \Sigma \) and \( A \in V \) and \( A \rightarrow w \) is a rule, then
\[ uv \xrightarrow{A} uvw \]
--- pronounced "yields"

Def \( u \xrightarrow{*} v \) if \( \exists u_1, u_2, u_3, \ldots, u_k \) st. \( u \xrightarrow{u_1} u_2 \xrightarrow{u_3} \ldots \xrightarrow{u_k} v \)
--- pronounced "derives"

\[ \text{Ex: } S \xrightarrow{*} \text{O00111 because there are rules so that eventually } S \text{ can become O00111} \]

Def \( I G \) is a \( CFG \), then
\[ L(G) = \{ w \mid w \in \Sigma^* \text{ and } S \xrightarrow{*} w \} \]
where \( S \) is start variable of \( G \)

Def \( L \) is a context free language if \( L = L(G) \) for some \( CFG \) \( G \)

\[ L_3 = \{ w \mid w \text{ is a grammatically correct English sentence} \} \]

\[ \langle \text{SENTENCE} \rangle \rightarrow \langle \text{NOUN PHRASE} \rangle \langle \text{VERB PHRASE} \rangle \]. high level
\[ \langle \text{NOUN PHRASE} \rangle \rightarrow \langle \text{NOUN MODIFIERS} \rangle \langle \text{NOUN} \rangle \]
\[ \langle \text{VERB PHRASE} \rangle \rightarrow \langle \text{VERB} \rangle \langle \text{VERB OBJECT} \rangle \]

\[ L_4 = \{ w \mid w \text{ is a valid arithmetic expression? (no - or %) } \}
\[ s \rightarrow n \mid \text{sum} \mid \text{product} \]
\[ n \rightarrow \{ 0 \mid 1 \mid \ldots \mid u9 \}^+ \]
\[ \text{sum} \rightarrow (s + s) \]
\[ \text{product} \rightarrow (s * s) \]
--- example 2.4 in book has something similar w/o parentheses
DO CFL include regular languages? Yes.

**Theorem:** If $L$ is regular, then it is also a CFL.

**Proof:** Let $R$ be a regex for $L$. Proof by strong induction on length of $R$.

**Base case:** $R = a$ for $a \in \Sigma$, $R = \epsilon$, $R = \emptyset$

**Base case:**
1. $S \rightarrow a$
2. $S \rightarrow \epsilon$
3. $S \rightarrow S$

**Inductive step:**
4. $R = (R_1) \cup (R_2)$
   - By induction, grammar for $R_1$, $R_2$, $S_1$, start state $S_1$
   - Grammar for $R_2$, $S_2$, start state $S_2$
   - Grammar for $R$:
     $$S \rightarrow S_1 \mid S_2$$
     - for $R = R_1 \cup R_2$
     $$S \rightarrow S_1, S_2$$
   - for $R = R_1^*$
     $$S \rightarrow \epsilon \mid S, S$$

**Theorem 2:** CFL are closed under $\cup$, $\circ$, $\ast$

By inductive step in the previous proof

Q: Are they also closed under complement and $\emptyset$?

A: No!

$$0^n 1^n 2^n$$ not a CFL, but

$$0^n 1^n 2^n = 0^n 1^n 2^* \cup 0^* 1^n 2^n$$
- **Chomsky Normal Form**

  In Chomsky form, rules are

  \[ A \rightarrow a \quad B \rightarrow B \]

  any string of terminals and nonterminals, \((V \cup \Sigma)^*\)

  **In Chomsky Normal Form**

  \[ A \rightarrow a \quad \text{terminal if } a \in \Sigma \]

  \[ A \rightarrow BC \quad \text{exactly two variables} \]

  \[ S \rightarrow \varepsilon \quad \text{is also allowed} \]

  \[ L_1 \text{ in Chomsky Normal Form} \]

  \[ L_1 = \{ \text{0 and 1} \} \]

\[
S_0 \rightarrow \varepsilon \\
S_0 \rightarrow \text{Zero A} \\
A \rightarrow \text{One} \\
S \rightarrow \text{Zero B} \\
B \rightarrow \text{One} \\
S \rightarrow \text{Zero One}
\]