Chapter 4: Grammars and Parsing

Context-Free Grammars:
- Terminal Vocabulary: \((\text{Terminals, } V_t)\)
- Intermediate symbols: \((\text{non-terminals, variables, } V_n)\)
- Start symbol (non-terminal) \(S\)
- Productions of the form \(A \rightarrow X_1 \, X_2 \, \ldots \, X_m\)
  - \(A\) is a non-terminal
  - \(X_1\) is a terminal or a non-terminal
  - valid: \(A \rightarrow\)

- Start with \(S\) and rewrite using productions until all terminals
  - Derivation: \(A \Rightarrow U_1 \ldots U_n \Rightarrow \ldots \Rightarrow a_1 \ldots a_m\)
  - Sentential form: Any "string" along the derivation
  - Sentence: final form with no non-terminals
Conventions:
- $a, b, c \rightarrow V_t$
- $A, B, C \rightarrow V_n$
- $U, V, W \rightarrow V \ (V_t + V_n)$
- $\text{Alpha, Beta, Gamma} \rightarrow V^*$
- $u, v, w \rightarrow V_t^*$

Leftmost derivation:
- $A \Rightarrow_{lm} X_1 \ X_2 \ldots \ X_m \Rightarrow_{lm} Y_1 \ldots \ Y_n \ X_2 \ldots \ X_m$
- All steps replace the left most non-terminal

Rightmost derivation
- $X_1 \ X_2 \ldots \ X_m \Rightarrow_{rm} X_1 \ldots \ X_{m-1} \ Y_1 \ldots \ Y_n$
- All steps replace the right most non-terminal
Parse tree:

- A parse tree ... tree form of derivation
  - A parse tree many have lots of derivations.
  - Each parse tree has one leftmost derivation
  - Each parse tree has one rightmost derivation
  - Some sentences have more than one parse tree
    - Grammars like this are ambiguous .... BAD grammar!
    - Also have more than one leftmost and rightmost derivations.

Language:

- Given grammar G, L(G) is the "language" generated by G.
- L(G) is the set of all possible terminal strings w where S =>* w.
What is a PARSE?

Given a grammar and a string x, answer the question(s)

- Is x a member of L(G)?
- If so, what is its parse tree?

Other grammars and issues:

- Regular Grammar: A -> a B or C -> d rules only

- Context Sensitive:
  - alpha A beta -> alpha delta beta

- Does a grammar generate the language you want?