Introduction to Language Theory and Compilation Exercises

Session 3: Introduction to grammars

Reminders

Grammars

A *grammar* is a quadruplet $G = \langle V, T, P, S\rangle$ where:

- V is a finite set of *variables*;
- T is a finite set of *terminals*;
- P is a finite set of *production rules* of the form $\alpha \to \beta$ with:
  - $\alpha \in (V \cup T)^* V (V \cup T)^*$ and
  - $\beta \in (V \cup T)^*$
- $S \in V$ is a variable called the *start symbol*.

Chomsky hierarchy

Class 0: Unrestricted grammars All grammars are in this class.

Class 1: Context-sensitive grammars A grammar $G = \langle V, T, P, S\rangle$ is *context sensitive* iff each rule $\alpha \to \beta \in P$ is s.t.:

1. either $\alpha = S$ and $\beta = \varepsilon$;
2. or $|\alpha| \leq |\beta|$ and S does not appear in β.

Class 2: Context-free grammars A grammar $G = \langle V, T, P, S\rangle$ is *context-free* iff each rule $\alpha \to \beta \in P$ is s.t.: $\alpha \in V$, i.e., the left-hand side is only one variable.

Class 3: Regular grammars A grammar $G = \langle V, T, P, S\rangle$ is *regular* iff it is *either* left-regular *or* right-regular:

Left-regular grammars G is left-regular iff each rule $\alpha \to \beta \in P$ is s.t. $\alpha \in V$ and either $\beta \in T^*$, or $\beta \in V \cdot T^*$.

Right-regular grammars G is left-regular iff each rule $\alpha \to \beta \in P$ is s.t. $\alpha \in V$ and either $\beta \in T^*$, or $\beta \in T^* \cdot V$.

Derivations

Let $G = \langle V, T, P, S\rangle$ be a grammar, and let γ and δ be s.t. $\gamma \in (V \cup T)^* V (V \cup T)^*$, and $\delta \in (V \cup T)^*$. Then, we say that δ *can be derived from* γ *(under the rules of* G*)*, written:

$$\gamma \Rightarrow_G \delta$$

iff there are $\gamma_1, \gamma_2 \in (V \cup T)^*$ and a rule $\alpha \to \beta \in P$ s.t.: $\gamma = \gamma_1 \cdot \alpha \cdot \gamma_2$ and $\delta = \gamma_1 \cdot \beta \cdot \gamma_2$.

The *language of* G is $L(G) = \{w \in T^* \mid S \Rightarrow_G^* w\}$ where $\Rightarrow_G^*$ is the reflexive and transitive closure of $\Rightarrow_G$.

For **context-free** grammars, a derivation $wSw' \Rightarrow_G w\alpha w'$, obtained by a applying $S \to \alpha$ is *left-most* iff $w \in T^*$. It is *rightmost* iff $w' \in T^*$.

Exercises

Ex. 1. Informally describe the languages generated by the following grammars and also specify what kind of grammars they are:

$$S \rightarrow 0 \mid 1 \mid 1S$$

(a): Grammar G_1

$$S \rightarrow a \mid *SS \mid +SS$$

(b): Grammar G_2

$$\begin{array}{rcl} S & \rightarrow & abcA \\ S & \rightarrow & Aabc \\ A & \rightarrow & \varepsilon \\ Aa & \rightarrow & Sa \\ cA & \rightarrow & cS \end{array}$$

(c): Grammar G_3

Give a derivation of the word 1110 produced by grammar G_1, a derivation of the word $* + a + aa * aa$ produced by grammar G_2 and a derivation of the word $abcabc$ produced by grammar G_3.

Ex. 2. Let G be the grammar in Figure 1.

1. To which class of grammars does G belong?
2. Give derivations of the following sentential forms in the form of a tree (with root labelled by S):
   a) $baSb$
   b) $bBABb$
   c) $baabaab$
3. Give the *leftmost* and *rightmost derivations* for $baabaab$.

$$\begin{array}{lcl} S & \rightarrow & AB \\ A & \rightarrow & Aa \\ A & \rightarrow & bB \\ B & \rightarrow & a \\ B & \rightarrow & Sb \end{array}$$

Figure 1: Grammar G

Ex. 3. Write a context-free grammar that generates all strings of as and bs (in any order) such that there are strictly more as than bs. Test your grammar on the input $baaba$ by giving a derivation.

Ex. 4. Write a context-sensitive grammar that generates all strings of as, bs and cs (in any order) such that there are as many of each. Give a derivation of $cacbab$ using your grammar.

(*Bonus*) Do you think such language can be generated by a context-free grammar? Informally explain why.