Operator Precedence Relations

Def: A context free grammar is an operator precedence grammar if

1. all right hand sides are unique with respect to terminal symbols,
2. there are no empty right hand sides,
3. no right hand side has two non-terminal symbols next to each other, and
4. at most one operator precedence relation holds between pairs of terminal symbols in the grammar.

The first three conditions are easily verifiable by simply inspecting the grammar. The fourth condition requires a little work. There are three operator precedence relations that hold between pairs of terminal symbols in a grammar: less than ($<$), equals ($=$), and greater than ($>$). They are defined as

Def: $V_i \leq_o V_j \text{ if } \exists$ a right hand side of a production of the form $\alpha V_i V_j \beta$ where $V_i, V_j \in V$ and $\alpha, \beta \in V'$

or

$\exists$ a right hand side of a production of the form $\alpha V_i N V_j \beta$ where $V_i, V_j \in V$, $N \in V_n \cup \{ \lambda \}$, and $\alpha, \beta \in V'$.

Def: $V_i <_o V_j \text{ if } \exists$ a right hand side of a production of the form $\alpha V_i V_k \beta$ and $V_k \Rightarrow^* V_j \delta$ where $V_k \in V_n$, $V_i, V_j \in V$, and $\alpha, \beta, \delta \in V'$

or

$\exists$ a right hand side of a production of the form $\alpha V_i V_k \beta$ and $V_k \Rightarrow^* N V_j \delta$ where $V_k \in V_n$, $V_i, V_j \in V$, $N \in V_n \cup \{ \lambda \}$, and $\alpha, \beta, \delta \in V'$.

Def: $V_i >_o V_j \text{ if } \exists$ a right hand side of a production of the form $\alpha V_k V_j \beta$ and $V_k \Rightarrow^* \delta V_i$

or

$\exists$ a right hand side of a production of the form $\alpha V_k V_l \beta$ and $V_k \Rightarrow^* \delta V_j$ and $V_l \Rightarrow^* V_j \gamma$ where $V_k, V_l \in V_n$, $V_i, V_j \in V$, and $\alpha, \beta, \delta, \gamma \in V'$

or

$\exists$ a right hand side of a production of the form $\alpha V_k V_j \beta$ and $V_k \Rightarrow^* \delta V_i N$, where $V_i, V_j \in V$, $N \in V_n \cup \{ \lambda \}$, and $\alpha, \beta, \delta \in V'$.

Let's also define two binary relations that will be helpful in computing operator precedence relations later.

Def: $V_i h_o V_j$ holds if $V_i \rightarrow V_j \alpha \in P$ where $V_i \in V_n$, $V_j \in V_n$, and $\alpha \in V'$

or

$V_i \rightarrow N V_j \alpha \in P$ where $V_i \in V_n$, $V_j \in V_t$, $N \in V_n \cup \{ \lambda \}$, and $\alpha \in V'$.

$h_o$ is called the immediate operator precedence head symbol relation. If $V_i h_o V_j$ holds, then we say that $V_i$ has an immediate operator precedence head symbol $V_j$. Furthermore let the binary relation $H_o$ be defined as
Def: $V_i \Rightarrow^* V_j \alpha$ where $V_i \in V_n$, $V_j \in V_n$, and $\alpha \in V^*$
or $V_i \Rightarrow^* N V_j \alpha$ where $V_i \in V_n$, $V_j \in V_n$, $N \in V_n \cup \{ \lambda \}$, and $\alpha \in V^*$.

$H_0$ is called the operator precedence head symbol relation. If $V_i H_0 V_j$ holds, then we say that $V_i$ has an operator precedence head symbol $V_j$. Note that $H_0$ is the transitive closure of $h_o$.

$$H_0 = h_o^*$$

In a similar fashion, we can define

Def: $V_i t_o V_j$ holds if $V_i \rightarrow \alpha$ $V_j \in P$ where $V_i \in V_n$, $V_j \in V_n$, and $\alpha \in V^*$
or $V_i \rightarrow \alpha$ $V_j \in P$ where $V_i \in V_n$, $V_j \in V_n$, $N \in V_n \cup \{ \lambda \}$, and $\alpha \in V^*$.

$t_o$ is called the immediate operator precedence tail symbol relation. If $V_i t_o V_j$ holds, then we say that $V_i$ has an immediate operator precedence tail symbol $V_j$. Furthermore let the binary relation $T_o$ be defined as

Def: $V_i T_o V_j$ holds if $V_i \Rightarrow^* \alpha$ $V_j \in N$ where $V_i \in V_n$, $V_j \in V_n$, and $\alpha \in V^*$
or $V_i \Rightarrow^* \alpha$ $V_j \in N$ where $V_i \in V_n$, $V_j \in V_n$, $N \in V_n \cup \{ \lambda \}$, and $\alpha \in V^*$.

$T_o$ is called the operator precedence tail symbol relation. If $V_i T_o V_j$ holds, then we say that $V_i$ has an operator precedence tail symbol $V_j$. Note that $T_o$ is the transitive closure of $t_o$.

$$T_o = t_o^*$$

Next let us define a binary relation for the equals operator precedence relation by

Def: $V_i E_o V_j$ if $\exists$ a right hand side of a production of the form $\alpha V_i V_j \beta$ where $V_i$, $V_j \in V$ and $\alpha$, $\beta \in V^*$
or $\exists$ a right hand side of a production of the form $\alpha V_i N V_j \beta$ where $V_i$, $V_j \in V$, $N \in V_n \cup \{ \lambda \}$, and $\alpha$, $\beta \in V^*$.

If $V_i E_o V_j$ holds, then we say that $V_i$ can be next to (equals) $V_j$ in some right hand side of a production in an operator precedence grammar. Note that the operator precedence equals is similar to the simple precedence equals except that the relation only holds between pairs of terminal symbols and you "look through" and non-terminals on the right hand side to observe the equals relations. Note that by inspecting the grammar, we can identify all $E_o$ relations that hold.

The less than operator precedence relation can be identified as

Def: $V_i L_o V_j$ if $\exists$ a right hand side of a production of the form $\alpha V_i V_k \beta \delta$ where $V_k \in V_n$, $V_i$, $V_j \in V$, and $\alpha$, $\beta$, $\delta \in V^*$
or $\exists$ a right hand side of a production of the form $\alpha V_i N V_k \beta \delta$ where $V_k \in V_n$, $V_i$, $V_j \in V$, $N \in V_n \cup \{ \lambda \}$, and $\alpha$, $\beta$, $\delta \in V^*$.
If $V_i \xrightarrow{L_o} V_j$ holds, then we say that $V_i$ is less than $V_j$ and that $V_j$ starts a phrase in a sentential form. Let's investigate the $L_o$ relation a little further. Note that I can write the binary relation in the following form:

$$V_i \xrightarrow{L_o} V_j \text{ holds } \iff V_i \xrightarrow{E_o} V_k \text{ and } V_k \xrightarrow{H_o} V_j$$

This is the same definition with the binary relations substituted for the English narrative. If the $E_o$ and $H_o$ are binary relations that are represented by Boolean matrices $E_o$ and $H_o$, respectively, then we have

$$L_o = E_o \circ H_o$$

via relational composition. Hence, if the equals relations are represented as a Boolean matrix and the head symbols are also represented as a Boolean matrix and if the row and columns are ordered identically, then the less than relations can be computed by multiplying the $E_o$ matrix with the $H_o$ matrix.

The **greater than** operator precedence relation can also be defined using binary relations as

**Def:** $V_i \xrightarrow{G_o} V_j$ if 

- $\exists$ a right hand side of a production of the form $\alpha \ V_k \ V_j \beta$ and $V_k \Rightarrow^+ \delta \ V_i$ or
- $\exists$ a right hand side of a production of the form $\alpha \ V_k \ V_j \beta$ and $V_k \Rightarrow^+ \delta \ V_i$ and $V_l \Rightarrow^+ \ V_j \gamma$ where $V_k, V_l \in V_n, V_i, V_j \in V$, and $\alpha, \beta, \delta, \gamma \in V^*$

- $\exists$ a right hand side of a production of the form $\alpha \ V_k \ V_j \beta$ and $V_k \Rightarrow^+ \delta \ V_i \ N$, where $V_i, V_j \in V_n \cup \{ \lambda \}$, and $\alpha, \beta, \delta \in V^*$.

If $V_i \xrightarrow{G_o} V_j$ holds, then we say that $V_i$ is greater than $V_j$ and that $V_i$ ends a phrase in a sentential form. We can write the binary relation for $G_o$ as

$$V_i \xrightarrow{G_o} V_j \text{ holds } \iff V_k \xrightarrow{E_o} V_j \text{ and } V_k \xrightarrow{T_o} V_i$$

Concentrating on $G_o$ for the moment, we note that the above form is not in the correct form for relational composition. However, by again re-writing the above definition we can get the definition into the correct form for relational composition, namely

$$V_i \xrightarrow{G_o} V_j \text{ holds } \iff V_k \xrightarrow{E_o} V_j \text{ and } V_i \xrightarrow{T_o^T} V_k$$

Again re-writing we have

$$V_i \xrightarrow{G_o} V_j \text{ holds } \iff V_i \xrightarrow{T_o^T} V_k \text{ and } V_k \xrightarrow{E_o} V_j$$

which is in the correct form for relational composition. If we have Boolean matrices $T_o$ and $E_o$ that represent the binary relations $T_o$ and $E_o$, respectively, then we have

$$G_o = T_o^T \ E_o$$
Review:

Computationally, we have

1. Identify $h_o$ by looking at the grammar
2. Compute $H_o$ by taking the transitive closure of $h_o$
3. Identify $t_o$ by looking at the grammar
4. Compute $T_o$ by taking the transitive closure of $t_o$
5. Identify $E_o$ by looking at the grammar
6. Compute $L_o$ by multiplying $E_o H_o$
7. Compute $G_o$ by multiplying $T_o^T E$