Tree Grammar

- **Regular tree grammar** : $G = (N, \mathcal{F}, R, S)$
  - N: a set of non-terminal symbols
  - $\mathcal{F}$: a set of terminal (function) symbols
  - R: a set of production rules

    Form of production rules

    $A \to \alpha \quad (A \in N,\ \alpha \in \mathsf{T}(\mathcal{F} \cup N))$
  - S ($\in N$): Start symbol (axiom)

- **Ex:** $G = (\{List, Nat\}, \{0, nil, s(), cons(,)\}, R, List)$, **where** R **consists of the following rules**

  $List \rightarrow nil$, $List \rightarrow cons(Nat, List)$,

  $Nat \rightarrow 0$, $Nat \rightarrow s(Nat)$

  - **Production of** $cons(s(0), nil)$

    $$\begin{aligned} & List \\ \rightarrow_G\ & cons(Nat, List) \\ \rightarrow_G\ & cons(s(Nat), List) \\ \rightarrow_G\ & cons(s(0), List) \\ \rightarrow_G\ & cons(s(0), nil) \end{aligned}$$

- **Derivation relation** $\rightarrow_G$ **(**$\subseteq \mathsf{T}(\mathcal{F} \cup N) \times \mathsf{T}(\mathcal{F} \cup N)$**):** **minimal set satisfying**
  - $R \subseteq \rightarrow_G$, **and**
  - $\alpha \rightarrow_G \beta$ **implies** $f(\cdots \alpha \cdots) \rightarrow_G f(\cdots \beta \cdots)$
- **Language generated by** G**:**

  $L(G) = \{s \mid S \rightarrow_G^* s \in \mathsf{T}(\mathcal{F})\}$

- **Theorem: A language is generated by regular tree grammar if and only if it is regular**
- **Proof ($\supseteq$): Construct regular tree grammar** $G = (Q \cup \{S\}, \mathcal{F}, R, S)$ **from NFTA** $A = (Q, \mathcal{F}, Q^f, \Delta)$**:**
  - $S \to q \in R$ **for each** $q \in Q^f$**, and**
  - $q \to f(q_1, \ldots, q_n) \in R$ **for each** $f(q_1, \ldots, q_n) \to q$
- **Proof sketch ($\subseteq$): Construct NFTA** $A = (N \cup Q, \mathcal{F}, \{S\}, \Delta)$ **from regular tree grammar** $G = (N, \mathcal{F}, R, S)$**:**
  - **For each** $List \to cons(s(Nat), List) \in R$**, construct rules by introducing a fresh state** q**:**

$$s(Nat) \to q \in \Delta, \quad cons(q, List) \to List \in \Delta$$

- **Q5-1: Let $\mathcal{F} = \{a, g(), f(,)\}$. Show a regular tree grammar equivalent to following NFTA:**

$$\begin{array}{rclrcl} a & \to & q_a & a & \to & q_\perp \\ f(q_a, q_\perp) & \to & q_1 & g(q_\perp) & \to & q_\perp \\ g(q_1) & \to & q_\varepsilon & f(q_\perp, q_\perp) & \to & q_\perp \end{array}$$

- **Q5-2: Let $\mathcal{F} = \{a, g(), f(,)\}$. Show an NFTA with start symbol X that is equivalent to the following grammar:**

  $X \to f(g(A), A)$**,** $\quad A \to g(g(A))$**,** $\quad A \to a$

- **Q5-3: Complete the proof ($\supseteq$) in page 4, i.e. prove by induction on n that $t \to_A^n q$ implies $q \to_G^* t$ for any $t \in \top(\mathcal{F})$, $q \in Q$, $n \in N$**

Regular tree expressions

- Ex.: regular tree expression $f(\square, \square)^{*,\square} \;._\square\; a$ represents language $\mathsf{T}(\{f(,), a\})$
- Ex.: regular tree expression $s(\square)^{*,\square} \;._\square\; 0$ represents language $\{s^n(0) \mid n \geq 0\}$
- Ex.: regular tree expression $cons((s(\square_1)^{*,\square_1} \;._{\square_1}\; 0), \square_2)^{*,\square_2} \;._{\square_2}\; nil$ represents 'List'

- $\mathcal{K} = \{\Box_1, \ldots\}$ **is set of variables for substitution**
- **tree substitution: (Rem.: different notion from substitution in slide 1)**

  For $t \in \mathsf{T}(\mathcal{F} \cup \mathcal{K})$ **and a tree language** L_i**,**

$$t\{\Box_1 \leftarrow L_1, \ldots, \Box_n \leftarrow L_n\} = \begin{cases} L_i & \cdots \textbf{if } t = \Box_i \\ \{a\} & \cdots \textbf{if } t = a \in \mathcal{F} \\ \{f(t_1, \ldots, t_n) \mid t_i \in s_i\{\Box_1 \leftarrow L_1, \ldots, \Box_n \leftarrow L_n\}\} & \\ & \cdots \textbf{if } t = f(s_1, \ldots, s_n) \end{cases}$$

$$L\{\Box_1 \leftarrow L_1, \ldots, \Box_n \leftarrow L_n\} = \bigcup_{t \in L} t\{\Box_1 \leftarrow L_1, \ldots, \Box_n \leftarrow L_n\}$$

- **Ex.:** $f(\Box, \Box)\{\Box \leftarrow \{a, b\}\} = \{f(a,a), f(a,b), f(b,a), f(b,b)\}$

- **Operations on languages**
  - **Concatenation:** $L \cdot_{\Box_i} M = L\{\Box_i \to M\}$
  - **Power:**
    $$L^{0,\Box_i} = \{\Box_i\}, \quad L^{k+1,\Box_i} = L^{k,\Box_i} \cup L \cdot_{\Box_i} L^{k,\Box_i}$$
  - **Closure:** $L^{*,\Box_i} = \bigcup_{k \geq 0} L^{k,\Box_i}$
- **Ex.:**

$$\{a, f(\Box,\Box)\}^{*,\Box} =$$
$$\{\Box\} \cup$$
$$\{a, f(\Box,\Box)\} \cup$$
$$\{f(\Box,a), f(\Box,f(\Box,\Box)),$$
$$f(a,\Box), f(a,a), f(a,f(\Box,\Box)),$$
$$f(f(\Box,\Box),\Box), f(f(\Box,\Box),a), f(f(\Box,\Box),f(\Box,\Box))\} \cup$$
$$\vdots$$

- **Regular tree expression** E, **tree language** $[\![E]\!]$**:**
  Let E **and** E_i **be regular tree expressions (RTE)**
  - $\emptyset$ **is RTE, and** $[\![\emptyset]\!] = \{\}$
  - $a \in \mathcal{F} \cup \mathcal{K}$ **is RTE, and** $[\![a]\!] = \{a\}$
  - **For** $f \in \mathcal{F}$**,** $f(E_1, \dots, E_n)$ **is RTE, and**
    $[\![f(E_1, \dots, E_n)]\!] = \{f(t_1, \dots, t_n) \mid t_i \in [\![E_1]\!]\}$
  - $E_1 + E_2$ **is RTE, and** $[\![E_1 + E_2]\!] = [\![E_1]\!] \cup [\![E_2]\!]$
  - $E_1 \cdot_{\Box_i} E_2$ **is RTE, and**
    $[\![E_1 \cdot_{\Box_i} E_2]\!] = [\![E_1]\!] \cdot_{\Box_i} [\![E_2]\!]$
  - $E^{*,\Box_i}$ **is RTE, and** $[\![E^{*,\Box_i}]\!] = [\![E]\!]^{*,\Box_i}$

- **Q5-4: Consider languages on** $\mathcal{F} = \{a, g(), f(,)\}$**:**
  $L_1 = \{g^k(a) \mid k \geq 0\}$**, and**
  $L_2 = \{f(t_1, t_2) \mid t_1, t_2 \in L_1\}$
  Show a regular tree expression that represents $L_1 \cup L_2$