F12

Lambda calculus

Lennart Andersson

Revision 2009-04-28

2009

Lambda calculus grammars

Concrete grammar

$$term ::= id \mid (\lambda\, id\, .\, term) \mid (term\ term)$$

Haskell representation

```
data Lambda = Id String
            | Abstr String Lambda
            | Appl Lambda Lambda
```

Abstract grammar for **Lexp** (Nielsen style)

$$M ::= x \mid \lambda x\,.\,M \mid M_1\ M_2$$

Simplifed concrete syntax

Applications associate to the left

$$M_1\ M_2\ M_3\ \ldots\ M_n \triangleq (\ldots((M_1\ M_2)\ M_3)\ \ldots\ M_n)$$

Abstractions associate to the right

$$\lambda\sigma_1\,.\,\lambda\sigma_2\,.\ldots.\,\lambda\sigma_n\,.\,M \triangleq (\lambda\sigma_1.(\lambda\sigma_2\,.(\ldots\ (\lambda\sigma_n\,.\,M)\ldots)))$$

Abstractions extends as far as possible

$$\lambda\sigma\,.\,M_1\ M_2\ M_3\ \ldots\ M_n \triangleq \lambda\sigma.(M_1\ M_2\ M_3\ \ldots\ M_n)$$

Free and bound identifiers

$$\mathcal{F}(\sigma) \triangleq \{\,\sigma\,\}$$
$$\mathcal{F}(\lambda\sigma\,.\,M) \triangleq \mathcal{F}(M) \setminus \{\,\sigma\,\}$$
$$\mathcal{F}(M_1\ M_2) \triangleq \mathcal{F}(M_1) \cup \mathcal{F}(M_2)$$

$$\mathcal{B}(\sigma) \triangleq \emptyset$$
$$\mathcal{B}(\lambda\sigma\,.\,M) \triangleq \mathcal{B}(M) \cup \{\,\sigma\,\}$$
$$\mathcal{B}(M_1\ M_2) \triangleq \mathcal{B}(M_1) \cup \mathcal{B}(M_2)$$

$$\mathcal{I}(M) \triangleq \mathcal{F}(M) \cup \mathcal{B}(M)$$

Combinators

$$I \triangleq \lambda x\,.\,x$$
$$K \triangleq \lambda x\,.\,\lambda y\,.\,x$$
$$K_* \triangleq \lambda x\,.\,\lambda y\,.\,y$$
$$S \triangleq \lambda x\,.\,\lambda y\,.\,\lambda z\,.\,xz(yz)$$

Substitution ...

$$\sigma[\sigma \mapsto M] \triangleq M$$
$$\tau[\sigma \mapsto M] \triangleq \tau \qquad \text{if } \tau \text{ and } \sigma \text{ are different identifiers}$$
$$(M_1 M_2)[\sigma \mapsto M] \triangleq (M_1[\sigma \mapsto M])(M_2[\sigma \mapsto M])$$

... Substitution

$$(\lambda\sigma\,.\,M)[\sigma \mapsto N] \triangleq (\lambda\sigma\,.\,M)$$
$$(\lambda\tau\,.\,M)[\sigma \mapsto N] \triangleq (\lambda\tau\,.\,(M[\sigma \mapsto N])) \quad \text{if } \tau \neq \sigma \wedge \tau \notin \mathcal{F}(N)$$
$$(\lambda\tau\,.\,M)[\sigma \mapsto N] \triangleq (\lambda\upsilon\,.\,((M[\tau \mapsto \upsilon])[\sigma \mapsto N])) \quad \text{if } \tau \neq \sigma \wedge \tau \in \mathcal{F}(N) \wedge \upsilon \notin \mathcal{I}(M) \cup \mathcal{I}(N) \cup \{\sigma\}$$

Church-Rosser theorem

Theorem. *If $M \twoheadrightarrow_\beta N$ and $M \twoheadrightarrow_\beta N'$ where N are N' normal forms then $N =_\alpha N'$.* ■

Fixed point combinator

Theorem. *For any combinator F there is a combinator X such that $F\,X = X$. Furthermore, there is a combinator, $Y \triangleq \lambda f.(\lambda x\,.\,f(x\;x))(\lambda x\,.\;f(x\;x))$, such that $Y\;F$ is a fixed point of F, i.e. $Y\;F = F(Y\;F)$.* ■