

(Atom Lenses) $al ::= Iterate(dl) \mid Identity$
 (Clause Lenses) $cl ::= ([al_1; \dots; al_n], [s_{1,1}; \dots; s_{1,n+1}], [s_{2,1}; \dots; s_{2,n+1}], \sigma \in [0, n] \mapsto [0, n])$
 (DNF Lenses) $dl ::= ([cll_1; \dots; cll_n], \sigma \in [0, n] \mapsto [0, n])$

Figure 1: Regex Syntax

Underlying Function of DNF Lenses:

- $[cll_1; \dots; cll_n].putr(s) = cll_1(s)$ if $s \in dom(cll_1)$ or ... or $cll_n(s)$ if $s \in dom(cll_n)$

Underlying Function of Clause Lenses:

- $([al_1; \dots; al_n], [s_{1,1}; \dots; s_{1,n+1}], [s_{2,1}; \dots; s_{2,n+1}]).putr(s_{1,1}.s_1 \dots s_n.s_{1,n+1}) = s_{2,1}.(al_{\sigma^{-1}(1)}.putr(s_{\sigma^{-1}(1)})) \dots (al_{\sigma^{-1}(n)}.putr(s_{\sigma^{-1}(n)})).s_{2,n+1}$

Underlying Function of Atom Lenses:

- $identity.putr(s) = s$
- $iterate(l).putr(\epsilon) = \epsilon$
- $iterate(l).putr(s.s') = (l.putr(s)).(iterate(l).putr(s'))$