Recall Quick Sort

```
0 : Nat
s : Nat → Nat
nil : List a
cons : a → List a → List a
false : Bool
true : Bool

not true → false
not false → true
lt x 0 → false
lt 0 (s y) → true
lt (s x) (s y) → lt x y
geq x y → not (lt x y)
append nil ys → ys
append (cons x xs) ys → cons x (append xs ys)
filter p nil → nil
filter p (cons x xs) → if_filter p x xs (p x)
if_filter p x xs true → cons x (filter p xs)
if_filter p x xs false → filter p xs

qsort nil → nil
qsort (cons x xs) → append (qsort (filter (geq x) xs))
                           (cons x (qsort (filter (lt x) xs)))
reverse nil → nil
reverse (cons x xs) → append (reverse xs) (cons x nil)
shuffle nil → nil
shuffle (cons x xs) → cons x (reverse (shuffle xs))
range 0 → nil
range (s x) → cons x (range x)
main → qsort (shuffle (range (s 0)))
```


Inferred Types

$$
\begin{aligned}
\text{not} &: \text{Bool} \to \text{Bool} \\
\text{lt} &: \text{Nat} \to \text{Nat} \to \text{Bool} \\
\text{geq} &: \text{Nat} \to \text{Nat} \to \text{Bool} \\
\text{append} &: \text{List Nat} \to \text{List Nat} \to \text{List Nat} \\
\text{filter} &: (\text{Nat} \to \text{Bool}) \to \text{List Nat} \to \text{List Nat} \\
\text{if\_filter} &: (\text{Nat} \to \text{Bool}) \to \text{Nat} \to \text{List Nat} \to \text{Bool} \to \text{List Nat} \\
\text{qsort} &: \text{List Nat} \to \text{List Nat} \\
\text{reverse} &: \text{List Nat} \to \text{List Nat} \\
\text{shuffle} &: \text{List Nat} \to \text{List Nat} \\
\text{range} &: \text{Nat} \to \text{List Nat} \\
\text{main} &: \text{List Nat}
\end{aligned}
$$

Why Are Inferred Types Monomorphic?

$$
\Gamma = \left\{ \begin{array}{r} 0 : \text{Nat} \\ \text{nil} : \text{List } a \\ \text{cons} : a \to \text{List } a \to \text{List } a \end{array} \right\}
$$

$$
\mathcal{R} = \left\{ \begin{array}{r} \text{append nil } ys \to ys \\ \text{append (cons } x\ xs)\ ys \to \text{cons } x\ (\text{append } xs\ ys) \\ \text{main} \to \text{append nil (cons 0 nil)} \end{array} \right\}
$$

mgu of $\mathcal{C}_\Gamma(\mathcal{R})$ yields

$$
\begin{aligned}
&\text{append} : \text{List Nat} \to \text{List Nat} \to \text{List Nat} \\
\text{but not} \quad &\text{append} : \text{List } a \to \text{List } a \to \text{List } a
\end{aligned}
$$

Solution

separate rules that define functions from those that use them

Type Inference with Polymorphism

$$
\Gamma_0 = \left\{ \begin{array}{l} \text{nil} : \text{List } a \\ \text{cons} : a \to \text{List } a \to \text{List } a \\ 0 : \text{Nat} \end{array} \right\}
$$

$$
\mathcal{R}_0 = \left\{ \begin{array}{l} \text{append nil } ys \to ys \\ \text{append (cons } x\ xs)\ ys \to \text{cons } x\ (\text{append } xs\ ys) \end{array} \right\}
$$

$\Gamma_1 = \Gamma_0 \cup \{\text{append} : \text{List } a \to \text{List } a \to \text{List } a\}$ by mgu of $\mathcal{C}_{\Gamma_0}(\mathcal{R}_0)$

$\mathcal{R}_1 = \{\ \text{main} \to \text{append nil (cons 0 nil})\ \}$

$\Gamma_2 = \Gamma_1 \cup \{\text{main} : \text{List nat}\}$ by mgu of $\mathcal{C}_{\Gamma_1}(\mathcal{R}_1)$

Question

how to determine groups of rules and order of type inference?

Answer

1. analyze dependencies of functions
2. compute strongly connected components (SCCs)
3. topologically sort them

Definition

- $\mathrm{head}(a\ \ell_1\ \cdots\ \ell_n) = a$, where a is constant or variable — head symbol
- $\mathcal{D}_\mathcal{R} = \{\mathrm{head}(\ell) \mid \ell \to r \in \mathcal{R}\}$ — defined symbols
- $\mathcal{R}_f = \{\ell \to r \in \mathcal{R} \mid \mathrm{head}(\ell) = f\}$ — f-rules
- $f \to g$ if some rule $\ell \to r$ in $\mathcal{R}_f$ contains $g \in \mathcal{D}_\mathcal{R}$ — dependency relation

$$\mathcal{R}_0 = \left\{ \begin{array}{l} \mathsf{not\ true} \to \mathsf{false} \\ \mathsf{not\ false} \to \mathsf{true} \end{array} \right\}$$

$$\mathcal{R}_1 = \left\{ \begin{array}{ll} \mathsf{lt}\ x\ 0 & \to \mathsf{false} \\ \mathsf{lt}\ 0\ (\mathsf{s}\ y) & \to \mathsf{true} \\ \mathsf{lt}\ (\mathsf{s}\ x)\ (\mathsf{s}\ y) & \to \mathsf{lt}\ x\ y \end{array} \right\}$$

$$\mathcal{R}_2 = \{\mathsf{geq}\ x\ y \to \mathsf{not}\ (\mathsf{lt}\ x\ y)\}$$

$$\mathcal{R}_3 = \left\{ \begin{array}{ll} \mathsf{append\ nil}\ ys & \to ys \\ \mathsf{append}\ (\mathsf{cons}\ x\ xs)\ ys & \to \mathsf{cons}\ x\ (\mathsf{append}\ xs\ ys) \end{array} \right\}$$

$$\mathcal{R}_4 = \left\{ \begin{array}{ll} \mathsf{filter}\ p\ \mathsf{nil} & \to \mathsf{nil} \\ \mathsf{filter}\ p\ (\mathsf{cons}\ x\ xs) & \to \mathsf{if\_filter}\ p\ x\ xs\ (p\ x) \\ \mathsf{if\_filter}\ p\ x\ xs\ \mathsf{true} & \to \mathsf{cons}\ x\ (\mathsf{filter}\ p\ xs) \\ \mathsf{if\_filter}\ p\ x\ xs\ \mathsf{false} & \to \mathsf{filter}\ p\ xs \end{array} \right\}$$

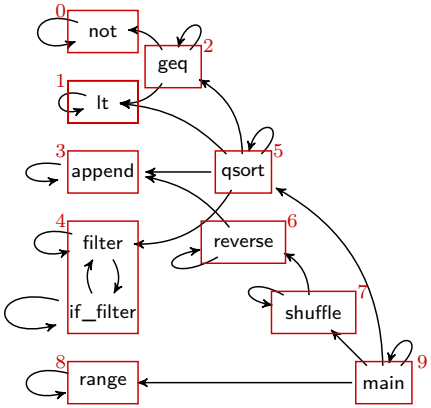
$$\mathcal{R}_5 = \left\{ \begin{array}{ll} \mathsf{qsort\ nil} & \to \mathsf{nil} \\ \mathsf{qsort}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{append}\ (\mathsf{qsort}\ (\mathsf{filter}\ (\mathsf{geq}\ x)\ xs))\ (\mathsf{cons}\ x\ (\mathsf{qsort}\ (\mathsf{filter}\ (\mathsf{lt}\ x)\ xs))) \end{array} \right\}$$

$$\mathcal{R}_6 = \left\{ \begin{array}{ll} \mathsf{reverse\ nil} & \to \mathsf{nil} \\ \mathsf{reverse}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{append}\ (\mathsf{reverse}\ xs)\ (\mathsf{cons}\ x\ \mathsf{nil}) \end{array} \right\}$$

$$\mathcal{R}_7 = \left\{ \begin{array}{ll} \mathsf{shuffle\ nil} & \to \mathsf{nil} \\ \mathsf{shuffle}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{cons}\ x\ (\mathsf{reverse}\ (\mathsf{shuffle}\ xs)) \end{array} \right\}$$

$$\mathcal{R}_8 = \left\{ \begin{array}{ll} \mathsf{range}\ 0 & \to \mathsf{nil} \\ \mathsf{range}\ (\mathsf{s}\ x) & \to \mathsf{cons}\ x\ (\mathsf{range}\ x) \end{array} \right\}$$

$$\mathcal{R}_9 = \{\mathsf{main} \to \mathsf{qsort}\ (\mathsf{shuffle}\ (\mathsf{range}\ (\mathsf{s}\ 0)))\}$$



$$\Gamma_0 = \left\{ \begin{array}{l} 0 : \mathsf{Nat} \\ \mathsf{s} : \mathsf{Nat} \to \mathsf{Nat} \\ \mathsf{nil} : \mathsf{List}\ a \\ \mathsf{cons} : a \to \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{false} : \mathsf{Bool} \\ \mathsf{true} : \mathsf{Bool} \end{array} \right\}$$

$$\mathcal{R}_0 = \left\{ \begin{array}{l} \mathsf{not\ true} \to \mathsf{false} \\ \mathsf{not\ false} \to \mathsf{true} \end{array} \right\}$$

$$\Gamma_1 = \Gamma_0 \cup \{\mathsf{not} : \mathsf{Bool} \to \mathsf{Bool}\}$$

$$\mathcal{R}_1 = \left\{ \begin{array}{ll} \mathsf{lt}\ x\ 0 & \to \mathsf{false} \\ \mathsf{lt}\ 0\ (\mathsf{s}\ y) & \to \mathsf{true} \\ \mathsf{lt}\ (\mathsf{s}\ x)\ (\mathsf{s}\ y) & \to \mathsf{lt}\ x\ y \end{array} \right\}$$

$$\Gamma_2 = \Gamma_1 \cup \{\mathsf{lt} : \mathsf{Nat} \to \mathsf{Nat} \to \mathsf{Bool}\}$$

$$\mathcal{R}_2 = \{\mathsf{geq}\ x\ y \to \mathsf{not}\ (\mathsf{lt}\ x\ y)\}$$

$$\Gamma_3 = \Gamma_2 \cup \{\mathsf{geq} : \mathsf{Nat} \to \mathsf{Nat} \to \mathsf{Bool}\}$$

$$\mathcal{R}_3 = \left\{ \begin{array}{ll} \mathsf{append\ nil}\ ys & \to ys \\ \mathsf{append}\ (\mathsf{cons}\ x\ xs)\ ys & \to \mathsf{cons}\ x\ (\mathsf{append}\ xs\ ys) \end{array} \right\}$$

$$\Gamma_4 = \Gamma_3 \cup \{\mathsf{append} : \mathsf{List}\ a \to \mathsf{List}\ a \to \mathsf{List}\ a\}$$

$$\mathcal{R}_4 = \left\{ \begin{array}{ll} \mathsf{filter}\ p\ \mathsf{nil} & \to \mathsf{nil} \\ \mathsf{filter}\ p\ (\mathsf{cons}\ x\ xs) & \to \mathsf{if\_filter}\ p\ x\ xs\ (p\ x) \\ \mathsf{if\_filter}\ p\ x\ xs\ \mathsf{true} & \to \mathsf{cons}\ x\ (\mathsf{filter}\ p\ xs) \\ \mathsf{if\_filter}\ p\ x\ xs\ \mathsf{false} & \to \mathsf{filter}\ p\ xs \end{array} \right\}$$

$$\Gamma_5 = \left\{ \begin{array}{l} \mathsf{filter} : (a \to \mathsf{Bool}) \to \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{if\_filter} : (a \to \mathsf{Bool}) \to a \to \mathsf{List}\ a \to \mathsf{Bool} \to \mathsf{List}\ a \end{array} \right\}$$

$$\mathcal{R}_5 = \left\{ \begin{array}{ll} \mathsf{qsort\ nil} & \to \mathsf{nil} \\ \mathsf{qsort}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{append}\ (\mathsf{qsort}\ (\mathsf{filter}\ (\mathsf{geq}\ x)\ xs))\ (\mathsf{cons}\ x\ (\mathsf{qsort}\ (\mathsf{filter}\ (\mathsf{lt}\ x)\ xs))) \end{array} \right\}$$

$$\Gamma_6 = \Gamma_5 \cup \{\mathsf{qsort} : \mathsf{List\ Nat} \to \mathsf{List\ Nat}\}$$

$$\mathcal{R}_6 = \left\{ \begin{array}{ll} \mathsf{reverse\ nil} & \to \mathsf{nil} \\ \mathsf{reverse}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{append}\ (\mathsf{reverse}\ xs)\ (\mathsf{cons}\ x\ \mathsf{nil}) \end{array} \right\}$$

$$\Gamma_7 = \Gamma_6 \cup \{\mathsf{reverse} : \mathsf{List}\ a \to \mathsf{List}\ a\}$$

$$\mathcal{R}_7 = \left\{ \begin{array}{ll} \mathsf{shuffle\ nil} & \to \mathsf{nil} \\ \mathsf{shuffle}\ (\mathsf{cons}\ x\ xs) & \to \mathsf{cons}\ x\ (\mathsf{reverse}\ (\mathsf{shuffle}\ xs)) \end{array} \right\}$$

$$\Gamma_8 = \Gamma_7 \cup \{\mathsf{shuffle} : \mathsf{List}\ a \to \mathsf{List}\ a\}$$

$$\mathcal{R}_8 = \left\{ \begin{array}{ll} \mathsf{range}\ 0 & \to \mathsf{nil} \\ \mathsf{range}\ (\mathsf{s}\ x) & \to \mathsf{cons}\ x\ (\mathsf{range}\ x) \end{array} \right\}$$

$$\Gamma_9 = \Gamma_8 \cup \{\mathsf{range} : \mathsf{Nat} \to \mathsf{List\ Nat}\}$$

$$\mathcal{R}_9 = \left\{ \mathsf{main} \to \mathsf{qsort}\ (\mathsf{shuffle}\ (\mathsf{range}\ (\mathsf{s}\ 0))) \right\}$$

$$\Gamma_{10} = \Gamma_9 \cup \{\mathsf{main} : \mathsf{List\ Nat}\}$$

Inferred Types

$$\begin{aligned} \mathsf{not} &: \mathsf{Bool} \to \mathsf{Bool} \\ \mathsf{lt} &: \mathsf{Nat} \to \mathsf{Nat} \to \mathsf{Bool} \\ \mathsf{geq} &: \mathsf{Nat} \to \mathsf{Nat} \to \mathsf{Bool} \\ \mathsf{append} &: \mathsf{List}\ a \to \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{filter} &: (a \to \mathsf{Bool}) \to \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{if\_filter} &: (a \to \mathsf{Bool}) \to a \to \mathsf{List}\ a \to \mathsf{Bool} \to \mathsf{List}\ a \\ \mathsf{qsort} &: \mathsf{List\ Nat} \to \mathsf{List\ Nat} \\ \mathsf{reverse} &: \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{shuffle} &: \mathsf{List}\ a \to \mathsf{List}\ a \\ \mathsf{range} &: \mathsf{Nat} \to \mathsf{List\ Nat} \\ \mathsf{main} &: \mathsf{List\ Nat} \end{aligned}$$