Problem 5

Problem 1. $\forall L \in \texttt{NatList}, \mathrm{size}(L) = \mathrm{size}(\mathrm{rev}(L))$.

Proof. By structural induction on L.

(1) Base case

What to show: $\mathrm{size}(nil) = \mathrm{size}(\mathrm{rev}(nil))$.

$$
\begin{aligned}
\underline{\mathrm{size}(nil)} &\longrightarrow 0 && \text{(by size1)}\\
\mathrm{size}(\underline{\mathrm{rev}(nil)}) &\longrightarrow \underline{\mathrm{size}(nil)} && \text{(by rev1)}\\
&\longrightarrow nil && \text{(by size1)}
\end{aligned}
$$

(2) Induction case

What to show: $\mathrm{size}(x \mid l) = \mathrm{size}(\mathrm{rev}(x \mid l))$

Induction hypothesis: $\mathrm{size}(l) = \mathrm{size}(\mathrm{rev}(l))$

where $x \in \texttt{PNat}$ and $l \in \texttt{NatList}$. Note that x, l are fresh constants[1].

$$
\begin{aligned}
\underline{\mathrm{size}(x \mid l)} &\longrightarrow \mathrm{s}(\underline{\mathrm{size}(l)}) && \text{(by size2)}\\
&\longrightarrow \underline{\mathrm{s}(\mathrm{size}(\mathrm{rev}(l)))} && \text{(by IH)}\\
&\longrightarrow \mathrm{size}(\mathrm{rev}(l)) + \mathrm{s}(0) && \text{(by Lemma 2)}\\
\mathrm{size}(\underline{\mathrm{rev}(x \mid l)}) &\longrightarrow \underline{\mathrm{size}(\mathrm{rev}(l) \mathbin{@} (x \mid nil))} && \text{(by rev2)}\\
&\longrightarrow \mathrm{size}(\mathrm{rev}(l)) + \underline{\mathrm{size}(x \mid nil)} && \text{(by Lemma 1)}\\
&\longrightarrow \mathrm{size}(\mathrm{rev}(l)) + \mathrm{s}(\underline{\mathrm{size}(nil)}) && \text{(by size2)}\\
&\longrightarrow \mathrm{size}(\mathrm{rev}(l)) + \mathrm{s}(0) && \text{(by size1)}
\end{aligned}
$$

□

Lemma 1. $\forall L1, L2 \in \texttt{NatList}, \mathrm{size}(L1 \mathbin{@} L2) = \mathrm{size}(L1) + \mathrm{size}(L1)$.

Proof. By structural induction on $L1$.

[1] A fresh constant of a sort denotes an arbitrary value of the sort, and has never been used before.

(1) Base case

What to show: $\mathrm{size}(nil \mathbin{@} l2) = \mathrm{size}(nil) + \mathrm{size}(l2)$
where $l2 \in$ `NatList`. Note that $l2$ is a fresh constant.

$$
\begin{aligned}
\mathrm{size}(\underline{nil \mathbin{@} l2}) &\longrightarrow \mathrm{size}(l2) && \text{(by @1)} \\
\underline{\mathrm{size}(nil)} \mathbin{@} \mathrm{size}(l2) &\longrightarrow \underline{0 + \mathrm{size}(l2)} && \text{(by size1)} \\
&\longrightarrow \mathrm{size}(l2) && \text{(by +1)}
\end{aligned}
$$

(2) Induction case

What to show: $\mathrm{size}((x \mid l1) \mathbin{@} l2) = \mathrm{size}(x \mid l1) + \mathrm{size}(l2)$
Induction hypothesis: $\mathrm{size}(l1 \mathbin{@} l2) = \mathrm{size}(l1) + \mathrm{size}(l2)$
where $x \in$ `PNat`, and $l1, l2 \in$ `NatList`. Note that $x, l1, l2$ are fresh constants.

$$
\begin{aligned}
\mathrm{size}(\underline{(x \mid l1) \mathbin{@} l2}) &\longrightarrow \underline{\mathrm{size}(x \mid (l1 \mathbin{@} l2))} && \text{(by @2)} \\
&\longrightarrow \mathrm{s}(\underline{\mathrm{size}(l1 \mathbin{@} l2)}) && \text{(by size2)} \\
&\longrightarrow \mathrm{s}(\mathrm{size}(l1) + \mathrm{size}(l2)) && \text{(by IH)} \\
\underline{\mathrm{size}(x \mid l1)} + \mathrm{size}(l2) &\longrightarrow \underline{\mathrm{s}(\mathrm{size}(l1)) + \mathrm{size}(l2)} && \text{(by size2)} \\
&\longrightarrow \mathrm{s}(\mathrm{size}(l1) + \mathrm{size}(l2)) && \text{(by +2)}
\end{aligned}
$$

□

Lemma 2. *$\forall X \in$ `PNat`$, \mathrm{s}(X) = X + \mathrm{s}(0)$.*

Proof. By structural induction on X.

(1) Base case

What to show: $\mathrm{s}(0) = 0 + \mathrm{s}(0)$.

$$
\underline{0 + \mathrm{s}(0)} \longrightarrow \mathrm{s}(0) \qquad \text{(by +1)}
$$

(2) Induction case

What to show: $\mathrm{s}(\mathrm{s}(x)) = \mathrm{s}(x) + \mathrm{s}(0)$
Induction hypothesis: $\mathrm{s}(x) = x + \mathrm{s}(0)$
where $x \in$ `PNat`. Note that x is a fresh constant.

$$
\begin{aligned}
\underline{\mathrm{s}(x) + \mathrm{s}(0)} &\longrightarrow \mathrm{s}(\underline{x + s(0)}) && \text{(by +2)} \\
&\longrightarrow \mathrm{s}(\underline{s(0) + x}) && \text{(by comm+)} \\
&\longrightarrow \mathrm{s}(s(\underline{0 + x})) && \text{(by +2)} \\
&\longrightarrow \mathrm{s}(\mathrm{s}(x)) && \text{(by +1)}
\end{aligned}
$$

□