MODULE *SumSequence*

This module contains a trivial *PlusCal* algorithm to sum the elements of a sequence of integers, together with its non-trivial complete *TLAPS*-checked proof.

This algorithm is one of the examples in Section 7.3 of "Proving Safety Properties", which is at

http://lamport.azurewebsites.net/tla/proving-safety.pdf

EXTENDS *Integers*, *SequenceTheorems*, *SequencesExtTheorems*, *NaturalsInduction*, *TLAPS*

To facilitate model checking, we assume that the sequence to be summed consists of integers in a set *Values* of integers.

CONSTANT *Values*

ASSUME $ValAssump \triangleq Values \subseteq Int$

In order to be able to express correctness of the algorithm, we define in TLA+ an operator *SeqSum* so that, if s is the sequence

$s\_1, \ \ldots\ , s\_n$

of integers, then $SumSeq(s)$ equals

$s\_1 + \ \ldots\ + s\_n$

The obvious TLA+ definition of *SeqSum* is

RECURSIVE $SeqSum(\_)$
$SeqSum(s) \triangleq$ IF $s = \langle\rangle$ THEN 0 ELSE $s[1] + SeqSum(Tail(s))$

However, *TLAPS* does not yet handle recursive operator definitions, but it does handle recursive function definitions. So, we define *SeqSum* in terms of a recursively defined function.

$SeqSum(s) \triangleq$
  LET $SS[ss \in Seq(Int)] \triangleq$ IF $ss = \langle\rangle$ THEN 0 ELSE $ss[1] + SS[Tail(ss)]$
  IN $SS[s]$

****

Here's the algorithm. It initially sets *seq* to an arbitrary sequence of integers in *Values* and leaves its value unchanged. It terminates with the variable sum equal to the sum of the elements of *seq*.

--fair algorithm $SumSequence$ **{**
  variables $seq \in Seq(Values)$, $sum = 0$, $n = 1$ **;**
  { a: **while (** $n \leq Len(seq)$ **)**
    { $sum := sum + seq[n]$ **;**
      $n := n + 1$ **;** **}**
  }
}

****

BEGIN TRANSLATION

VARIABLES *seq*, *sum*, *n*, *pc*

$vars \triangleq \langle seq, sum, n, pc \rangle$

$Init \triangleq$ Global variables
  $\land\ seq \in Seq(Values)$

$\land\ sum = 0$
$\land\ n = 1$
$\land\ pc = \text{"a"}$

$$
\begin{aligned}
a \triangleq\ & \land\ pc = \text{"a"} \\
& \land\ \text{IF } n \leq Len(seq) \\
& \qquad \text{THEN } \land\ sum' = sum + seq[n] \\
& \qquad \qquad\ \ \land\ n' = n + 1 \\
& \qquad \qquad\ \ \land\ pc' = \text{"a"} \\
& \qquad \text{ELSE } \land\ pc' = \text{"Done"} \\
& \qquad \qquad\ \ \land\ \text{UNCHANGED } \langle sum, n \rangle \\
& \land\ seq' = seq
\end{aligned}
$$

Allow infinite stuttering to prevent deadlock on termination.

$Terminating \triangleq pc = \text{"Done"} \land \text{UNCHANGED } vars$

$$
\begin{aligned}
Next \triangleq\ & a \\
& \lor\ Terminating
\end{aligned}
$$

$$
\begin{aligned}
Spec \triangleq\ & \land\ Init \land \Box[Next]_{vars} \\
& \land\ \text{WF}_{vars}(Next)
\end{aligned}
$$

$Termination \triangleq \Diamond(pc = \text{"Done"})$

END TRANSLATION

Correctness of the algorithm means that it satisfies these two properties:

- Safety: If it terminates, then it does so with sum equal to $SeqSum(seq)$.
- Liveness: The algorithm eventually terminates.

Safety is expressed in TLA+ by the invariance of the following postcondition.

$PCorrect \triangleq (pc = \text{"Done"}) \Rightarrow (sum = SeqSum(seq))$

To get TLC to check that the algorithm is correct, we use a model that overrides the definition of Seq so $Seq(S)$ is the set of sequences of elements of S having at most some small length. For example,

$Seq(S) \triangleq \text{UNION } \{[1 \,..\, i \rightarrow S] : i \in 0 \,..\, 3\}$

is the set of such sequences with length at most 3.

The Proof of Safety

To prove the invariance of the postcondition, we need to find an inductive invariant that implies it. A suitable inductive invariant is formula Inv defined here.

$$
\begin{aligned}
TypeOK \triangleq\ & \land\ seq \in Seq(Values) \\
& \land\ sum \in Int \\
& \land\ n \in 1 \,..\, (Len(seq) + 1) \\
& \land\ pc \in \{\text{"a"}, \text{"Done"}\}
\end{aligned}
$$

$$
\begin{aligned}
Inv \triangleq\ & \land\ TypeOK \\
& \land\ sum = SeqSum([i \in 1\,..\,(n-1) \mapsto seq[i]]) \\
& \land\ (pc = \text{“Done”}) \Rightarrow (n = Len(seq) + 1)
\end{aligned}
$$

TLC can check that *Inv* is an inductive invariant on a large enough model to give us confidence in its correctness. We can therefore try to use it to prove the postcondition.

In the course of writing the proof, I found that I needed two simple simple properties of sequences and *SeqSum*. The first essentially states that the definition of *SeqSum* is correct–that is, that it defines the operator we expect it to. TLA+ doesn't require you to prove anything when making a definition, and it allows you to write silly recursive definitions like

$$
\begin{aligned}
&\textsc{recursive}\ \ NotFactorial(\_) \\
&NotFactorial(i) \triangleq \textsc{if}\ \ i = 0\ \ \textsc{then}\ 1\ \ \textsc{else}\ \ i * NotFactorial(i+1)
\end{aligned}
$$

Writing this definition doesn't mean that $NonFactorial(4)$ actually equals $4 * NonFactorial(5)$. I think it actually does, but I'm not sure. I do know that it doesn't imply that $NonFactorial(4)$ is a natural number. But the recursive definition of *SeqSum* is sensible, and we can prove the following lemma, which implies that $SeqSum(\langle 1, 2, 3, 4\rangle)$ equals $1 + SeqSum(\langle 2, 3, 4\rangle)$.

$$
\begin{aligned}
&\textsc{lemma}\ Lemma1 \triangleq \\
&\quad \forall\, s \in Seq(Int) : \\
&\quad\quad SeqSum(s) = \textsc{if}\ \ s = \langle\rangle\ \ \textsc{then}\ 0\ \ \textsc{else}\ \ s[1] + SeqSum(Tail(s))
\end{aligned}
$$

What makes a formal proof of the algorithm non-trivial is that the definition of *SeqSum* essentially computes $SeqSum(seq)$ by summing the elements of *seq* from left to right, starting with $seq[1]$. However, the algorithm sums the elements from right to left, starting with $seq[Len(s)]$. Proving the correctness of the algorithm requires proving that the two ways of computing the sum produce the same result. To state that result, it's convenient to define the operator *Front* on sequences to be the mirror image of *Tail*:

$$Front(\langle 1, 2, 3, 4\rangle) = \langle 2, 3, 4\rangle$$

This operator is defined in the *SequenceTheorems* module. I find it more convenient to use the slightly different definition expressed by this theorem.

$$
\begin{aligned}
&\textsc{theorem}\ FrontDef \triangleq \forall\, S : \forall\, s \in Seq(S) : \\
&\qquad\qquad Front(s) = [i \in 1\,..\,(Len(s) - 1) \mapsto s[i]] \\
&\textsc{by}\ \ \textsc{def}\ Front
\end{aligned}
$$

$$
\begin{aligned}
&\textsc{lemma}\ Lemma5 \triangleq \forall\, s \in Seq(Int) : \\
&\qquad (Len(s) > 0) \Rightarrow \\
&\qquad\quad (SeqSum(s) = SeqSum(Front(s)) + s[Len(s)])
\end{aligned}
$$

If we're interested in correctness of an algorithm, we probably don't want to spend our time proving simple properties of data types. Instead of proving these two obviously correct lemmas, it's best to check them with *TLC* to make sure we haven't made some silly mistake in writing them, and to prove correctness of the algorithm. If we want to be sure that the lemmas are correct, we can then prove them. Proofs of these lemmas are given below.

$$
\begin{aligned}
&\textsc{theorem}\ Spec \Rightarrow \Box PCorrect \\
&\langle 1\rangle 1.\ Init \Rightarrow Inv \\
&\quad \langle 2\rangle\ \textsc{suffices assume}\ Init
\end{aligned}
$$

$\quad\quad\quad\quad$ PROVE $\quad$ Inv

OBVIOUS

$\langle 2 \rangle 1.\ TypeOK$

BY $Lemma1$, $ValAssump$ DEF $Init$, Inv, $TypeOK$

$\langle 2 \rangle 2.\ sum = SeqSum([i \in 1 \mathrel{..} (n-1) \mapsto seq[i]])$

$\langle 3 \rangle 1.\ (n-1) = 0$

BY DEF $Init$

$\langle 3 \rangle 2.\ [i \in 1 \mathrel{..} 0 \mapsto seq[i]] = \langle \rangle$

OBVIOUS

$\langle 3 \rangle 3.\ \langle \rangle \in Seq(Int)$

OBVIOUS

$\langle 3 \rangle 4.$ QED

BY $\langle 3 \rangle 2$, $\langle 3 \rangle 1$, $\langle 3 \rangle 3$, $Lemma1$ DEF $Init$

$\langle 2 \rangle 3.\ (pc = \text{"Done"}) \Rightarrow (n = Len(seq) + 1)$

BY $Lemma1$, $ValAssump$ DEF $Init$, Inv, $TypeOK$

$\langle 2 \rangle 4.$ QED

BY $\langle 2 \rangle 1$, $\langle 2 \rangle 2$, $\langle 2 \rangle 3$ DEF Inv

$\langle 1 \rangle 2.\ Inv \land [Next]_{vars} \Rightarrow Inv'$

$\langle 2 \rangle$ SUFFICES ASSUME Inv,

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ $[Next]_{vars}$

$\quad\quad\quad\quad$ PROVE $\quad$ Inv'

OBVIOUS

$\langle 2 \rangle$ USE $ValAssump$ DEF Inv, $TypeOK$

$\langle 2 \rangle 1.$CASE a

$\langle 3 \rangle 1.\ TypeOK'$

$\langle 4 \rangle 1.\ sum' \in Int$

$\langle 5 \rangle 1.$CASE $n \leq Len(seq)$

$\langle 6 \rangle .seq[n] \in Values$

BY $\langle 5 \rangle 1$

$\langle 6 \rangle .$QED BY $\langle 5 \rangle 1$, $\langle 2 \rangle 1$ DEF a

$\langle 5 \rangle 2.$CASE $\neg(n \leq Len(seq))$

BY $\langle 5 \rangle 2$, $\langle 2 \rangle 1$ DEF a

$\langle 5 \rangle .$QED BY $\langle 5 \rangle 1$, $\langle 5 \rangle 2$

$\langle 4 \rangle .$QED BY $\langle 4 \rangle 1$, $\langle 2 \rangle 1$ DEF a

$\langle 3 \rangle 2.\ (sum = SeqSum([i \in 1 \mathrel{..} (n-1) \mapsto seq[i]]))'$

$\langle 4 \rangle 1.$CASE $n > Len(seq)$

$\langle 5 \rangle\ \neg(n \leq Len(seq))$

BY $\langle 4 \rangle 1$ DEF Inv, $TypeOK$

$\langle 5 \rangle$ QED

BY $\langle 2 \rangle 1$, $\langle 4 \rangle 1$ DEF a, Inv, $TypeOK$

$\langle 4 \rangle 2.$CASE $n \in 1 \mathrel{..} Len(seq)$

$\langle 5 \rangle$ DEFINE $curseq \triangleq [i \in 1 \mathrel{..} (n-1) \mapsto seq[i]]$

$\quad\quad\quad\quad\quad\quad$ $s \triangleq curseq'$

$\langle 5 \rangle$ SUFFICES $sum' = SeqSum(s)$

OBVIOUS

$\langle 5 \rangle 1.\ \land n' - 1 = n$
$\quad \land Len(s) = n$
$\quad \land s[Len(s)] = seq[n]$
$\quad$ BY $\langle 2 \rangle 1, \langle 4 \rangle 2$ DEF $a, Inv, TypeOK$
$\langle 5 \rangle 2.\ s = [i \in 1 \,..\, n \mapsto seq[i]]$
$\quad$ BY $\langle 5 \rangle 1, \langle 2 \rangle 1$ DEF a
$\langle 5 \rangle 3.\ sum' = sum + seq[n]$
$\quad$ BY $\langle 2 \rangle 1, \langle 4 \rangle 2$ DEF a
$\langle 5 \rangle$ HIDE DEF s
$\langle 5 \rangle 4.\ SeqSum(s) = SeqSum([i \in 1 \,..\, (Len(s) - 1) \mapsto s[i]]) + s[Len(s)]$
$\langle 6 \rangle 1.\ \forall S, T : S \subseteq T \Rightarrow Seq(S) \subseteq Seq(T)$
$\quad$ OBVIOUS
$\langle 6 \rangle 2.\ seq \in Seq(Int)$
$\quad$ BY $\langle 6 \rangle 1, ValAssump$ DEF $Inv, TypeOK$
$\langle 6 \rangle 3.\ \forall i \in 1 \,..\, n : seq[i] \in Int$
$\quad$ BY $\langle 6 \rangle 2, \langle 4 \rangle 2$
$\langle 6 \rangle 4.\ s \in Seq(Int)$
$\quad$ BY $\langle 6 \rangle 3, \langle 5 \rangle 2, \langle 4 \rangle 2$
$\langle 6 \rangle 5.\ Front(s) = [i \in 1 \,..\, Len(s) - 1 \mapsto s[i]]$
$\quad$ BY $\langle 6 \rangle 4, FrontDef$
$\langle 6 \rangle$ QED
$\quad$ BY $\langle 6 \rangle 4, \langle 6 \rangle 5, \langle 5 \rangle 1, \langle 4 \rangle 2, Lemma5$
$\langle 5 \rangle 5.\ curseq = [i \in 1 \,..\, (Len(s) - 1) \mapsto s[i]]$
$\quad$ BY $\langle 5 \rangle 1, \langle 5 \rangle 2$
$\langle 5 \rangle 6.\ sum = SeqSum(curseq)$
$\quad$ BY $\langle 2 \rangle 1, \langle 4 \rangle 2, \langle 5 \rangle 5$ DEF $Inv, TypeOK, s$
$\langle 5 \rangle 7.$ QED
$\quad$ BY $\langle 5 \rangle 1, \langle 5 \rangle 3, \langle 5 \rangle 4, \langle 5 \rangle 5, \langle 5 \rangle 6$ DEF $Inv, TypeOK, s$
$\langle 4 \rangle 3.$ QED
$\quad$ BY $\langle 4 \rangle 1, \langle 4 \rangle 2$ DEF $Inv, TypeOK$
$\langle 3 \rangle 3.\ ((pc = \text{“Done”}) \Rightarrow (n = Len(seq) + 1))'$
$\quad$ BY $\langle 2 \rangle 1$ DEF $a, Inv, TypeOK$
$\langle 3 \rangle 4.$ QED
$\quad$ BY $\langle 3 \rangle 1, \langle 3 \rangle 2, \langle 3 \rangle 3$ DEF Inv
$\langle 2 \rangle 2.$CASE UNCHANGED $vars$
$\quad$ BY $\langle 2 \rangle 2$ DEF $Inv, TypeOK, vars$
$\langle 2 \rangle 3.$ QED
$\quad$ BY $\langle 2 \rangle 1, \langle 2 \rangle 2$ DEF $Next, Terminating$
$\langle 1 \rangle 3.\ Inv \Rightarrow PCorrect$
$\langle 2 \rangle$ SUFFICES ASSUME Inv,
$\quad pc = \text{“Done”}$
$\quad$ PROVE $sum = SeqSum(seq)$
$\quad$ BY DEF $PCorrect$
$\langle 2 \rangle 1.\ seq = [i \in 1 \,..\, Len(seq) \mapsto seq[i]]$
$\quad$ BY DEF $Inv, TypeOK$

$\langle 2\rangle 2.$ QED
  BY $\langle 2\rangle 1$ DEF Inv, $TypeOK$
$\langle 1\rangle 4.$ QED
  BY $\langle 1\rangle 1$, $\langle 1\rangle 2$, $\langle 1\rangle 3$, PTL DEF $Spec$

Proofs of the Lemmas.

The LET definition at the heart of the definition of *SeqSum* is a standard definition of a function on sequences by tail recursion. Theorem *TailInductiveDef* of module *SequenceTheorems* proves correctness of such a definition.

LEMMA $Lemma1\_Proof \triangleq$
$\quad \forall s \in Seq(Int) :$
$\quad\quad SeqSum(s) = \text{IF } s = \langle\rangle \text{ THEN } 0 \text{ ELSE } s[1] + SeqSum(Tail(s))$

$\langle 1\rangle$ DEFINE $DefSS(ssOfTailss, ss) \triangleq ss[1] + ssOfTailss$
$\quad SS[ss \in Seq(Int)] \triangleq$
$\quad\quad \text{IF } ss = \langle\rangle \text{ THEN } 0 \text{ ELSE } DefSS(SS[Tail(ss)], ss)$
$\langle 1\rangle 1.$ $TailInductiveDefHypothesis(SS, Int, 0, DefSS)$
  BY $Zenon$ DEF $TailInductiveDefHypothesis$
$\langle 1\rangle 2.$ $TailInductiveDefConclusion(SS, Int, 0, DefSS)$
  BY $\langle 1\rangle 1$, $TailInductiveDef$, $Zenon$
$\langle 1\rangle 3.$ $SS = [ss \in Seq(Int) \mapsto \text{IF } ss = \langle\rangle \text{ THEN } 0$
$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \text{ELSE } ss[1] + SS[Tail(ss)]]$
  BY $\langle 1\rangle 2$, $Zenon$ DEF $TailInductiveDefConclusion$
$\langle 1\rangle$ QED
  BY $\langle 1\rangle 3$, $Zenon$ DEF $SeqSum$

Lemmas 2 and 3 are simple properties of *Tail* and *Front* that are used in the proof of Lemma 5.

LEMMA $Lemma2 \triangleq$
$\quad \forall S : \forall s \in Seq(S) :$
$\quad\quad Len(s) > 0 \Rightarrow \land Tail(s) \in Seq(S)$
$\quad\quad\quad\quad\quad\quad\quad\quad \land Front(s) \in Seq(S)$
$\quad\quad\quad\quad\quad\quad\quad\quad \land Len(Tail(s)) = Len(s) - 1$
$\quad\quad\quad\quad\quad\quad\quad\quad \land Len(Front(s)) = Len(s) - 1$

$\langle 1\rangle$ SUFFICES ASSUME NEW S,
$\quad\quad\quad\quad$ NEW $s \in Seq(S)$,
$\quad\quad\quad\quad$ $Len(s) > 0$
$\quad\quad$ PROVE $\land Tail(s) \in Seq(S)$
$\quad\quad\quad\quad \land Front(s) \in Seq(S)$
$\quad\quad\quad\quad \land Len(Tail(s)) = Len(s) - 1$
$\quad\quad\quad\quad \land Len(Front(s)) = Len(s) - 1$
  OBVIOUS
$\langle 1\rangle 1.$ $Tail(s) \in Seq(S) \land Len(Tail(s)) = Len(s) - 1$
  OBVIOUS
$\langle 1\rangle 2.$ $Front(s) \in Seq(S) \land Len(Front(s)) = Len(s) - 1$
  BY $FrontDef$

$\langle 1\rangle 3.$ QED
  BY $\langle 1\rangle 1$, $\langle 1\rangle 2$

LEMMA $\mathit{Lemma2a} \triangleq$
  ASSUME NEW S, NEW $s \in \mathit{Seq}(S)$, $\mathit{Len}(s) > 1$
  PROVE $\mathit{Tail}(s) = [i \in 1 \,..\, (\mathit{Len}(s) - 1) \mapsto s[i+1]]$
$\langle 1\rangle$.DEFINE $t \triangleq [i \in 1 \,..\, (\mathit{Len}(s) - 1) \mapsto s[i+1]]$
$\langle 1\rangle 1.$ $\mathit{Tail}(s) \in \mathit{Seq}(S) \land t \in \mathit{Seq}(S)$
  OBVIOUS
$\langle 1\rangle 2.$ $\mathit{Len}(\mathit{Tail}(s)) = \mathit{Len}(t)$
  OBVIOUS
$\langle 1\rangle 3.$ $\forall i \in 1 \,..\, \mathit{Len}(\mathit{Tail}(s)) : \mathit{Tail}(s)[i] = t[i]$
  OBVIOUS
$\langle 1\rangle$.QED BY $\langle 1\rangle 1$, $\langle 1\rangle 2$, $\langle 1\rangle 3$

LEMMA $\mathit{Lemma3} \triangleq$
  $\forall S : \forall s \in \mathit{Seq}(S) :$
    $(\mathit{Len}(s) > 1) \Rightarrow (\mathit{Tail}(\mathit{Front}(s)) = \mathit{Front}(\mathit{Tail}(s)))$
  $\langle 1\rangle$ SUFFICES ASSUME NEW S,
      NEW $s \in \mathit{Seq}(S)$,
      $\mathit{Len}(s) > 1$
    PROVE $\mathit{Tail}(\mathit{Front}(s)) = \mathit{Front}(\mathit{Tail}(s))$
    OBVIOUS
  $\langle 1\rangle 1.$ $\mathit{Tail}(\mathit{Front}(s)) = [i \in 1 \,..\, (\mathit{Len}(s) - 2) \mapsto s[i+1]]$
    $\langle 2\rangle 1.$ $\land\ \mathit{Front}(s) = [i \in 1 \,..\, (\mathit{Len}(s) - 1) \mapsto s[i]]$
      $\land\ \mathit{Len}(\mathit{Front}(s)) = \mathit{Len}(s) - 1$
      $\land\ \mathit{Front}(s) \in \mathit{Seq}(S)$
      $\land\ \mathit{Len}(s) \in \mathit{Nat}$
      BY $\mathit{FrontDef}$
    $\langle 2\rangle 2.$ $\mathit{Len}(\mathit{Front}(s)) > 0$
      BY $\langle 2\rangle 1$
    $\langle 2\rangle 3.$ $\mathit{Front}(s) \neq \langle\rangle$
      BY $\langle 2\rangle 1$, $\langle 2\rangle 2$, Isa
    $\langle 2\rangle 4.$ $\mathit{Tail}(\mathit{Front}(s)) = [i \in 1 \,..\, (\mathit{Len}(\mathit{Front}(s)) - 1) \mapsto \mathit{Front}(s)[i+1]]$
      BY $\langle 2\rangle 1$, $\langle 2\rangle 3$, $\mathit{Lemma2a}$
    $\langle 2\rangle 5.$ $\forall i \in 0 \,..\, (\mathit{Len}(s) - 2) : \mathit{Front}(s)[i+1] = s[i+1]$
      BY $\langle 2\rangle 1$
    $\langle 2\rangle 6.$ $\mathit{Len}(\mathit{Front}(s)) - 1 = \mathit{Len}(s) - 2$
      BY $\langle 2\rangle 1$
    $\langle 2\rangle 7.$ $\mathit{Tail}(\mathit{Front}(s)) = [i \in 1 \,..\, (\mathit{Len}(s) - 2) \mapsto \mathit{Front}(s)[i+1]]$
      BY $\langle 2\rangle 4$, $\langle 2\rangle 6$
    $\langle 2\rangle 8.$ $\forall i \in 1 \,..\, (\mathit{Len}(s) - 2) : \mathit{Front}(s)[i+1] = s[i+1]$
      BY $\langle 2\rangle 5$, $\mathit{Z3}$
    $\langle 2\rangle 9.$ QED
      BY $\langle 2\rangle 7$, $\langle 2\rangle 8$

$\langle 1 \rangle 2.$ $\mathit{Front}(\mathit{Tail}(s)) = [i \in 1 \,..\, (\mathit{Len}(s) - 2) \mapsto s[i+1]]$
BY $\mathit{Len}(s) \in \mathit{Nat}$, $\mathit{Lemma2a}$ DEF Front
$\langle 1 \rangle 3.$ QED
BY $\langle 1 \rangle 1$, $\langle 1 \rangle 2$, Zenon

The following lemma asserts type correctness of the *SeqSum* operator. It's proved by induction on the length of its argument. Such simple induction is expressed by theorem *NatInduction* of module *NaturalsInduction*.

LEMMA $\mathit{Lemma4} \triangleq \forall s \in \mathit{Seq}(\mathit{Int}) : \mathit{SeqSum}(s) \in \mathit{Int}$
$\langle 1 \rangle$ DEFINE $P(N) \triangleq \forall s \in \mathit{Seq}(\mathit{Int}) : (\mathit{Len}(s) = N) \Rightarrow (\mathit{SeqSum}(s) \in \mathit{Int})$
$\langle 1 \rangle 1.$ $P(0)$
$\langle 2 \rangle$ SUFFICES ASSUME NEW $s \in \mathit{Seq}(\mathit{Int})$,
$\mathit{Len}(s) = 0$
PROVE $\mathit{SeqSum}(s) \in \mathit{Int}$
BY Zenon DEF P
$\langle 2 \rangle 1.$ $s = \langle\rangle$
OBVIOUS
$\langle 2 \rangle$ QED
BY $\langle 2 \rangle 1$, $\mathit{Lemma1}$, Isa
$\langle 1 \rangle 2.$ ASSUME NEW $N \in \mathit{Nat}$, $P(N)$
PROVE $P(N+1)$
$\langle 2 \rangle$ SUFFICES ASSUME NEW $s \in \mathit{Seq}(\mathit{Int})$,
$\mathit{Len}(s) = (N+1)$
PROVE $\mathit{SeqSum}(s) \in \mathit{Int}$
BY DEF P
$\langle 2 \rangle 1.$ $s \neq \langle\rangle$
OBVIOUS
$\langle 2 \rangle 2.$ $\mathit{SeqSum}(s) = s[1] + \mathit{SeqSum}(\mathit{Tail}(s))$
BY $\langle 2 \rangle 1$, $\mathit{Lemma1}$
$\langle 2 \rangle 3.$ $s[1] \in \mathit{Int}$
BY $\langle 2 \rangle 1$
$\langle 2 \rangle 4.$ $\land\ \mathit{Len}(\mathit{Tail}(s)) = N$
$\land\ \mathit{Tail}(s) \in \mathit{Seq}(\mathit{Int})$
BY $\langle 2 \rangle 2$, $\mathit{Lemma2}$
$\langle 2 \rangle 5.$ $\mathit{SeqSum}(\mathit{Tail}(s)) \in \mathit{Int}$
BY $\langle 1 \rangle 2$, $\langle 2 \rangle 4$, Zenon
$\langle 2 \rangle 6.$ QED
BY $\langle 2 \rangle 2$, $\langle 2 \rangle 3$, $\langle 2 \rangle 5$
$\langle 1 \rangle$ HIDE DEF P
$\langle 1 \rangle 3.$ $\forall N \in \mathit{Nat} : P(N)$
BY $\langle 1 \rangle 1$, $\langle 1 \rangle 2$, $\mathit{NatInduction}$, Isa
$\langle 1 \rangle 4.$ QED
BY $\langle 1 \rangle 3$ DEF P

LEMMA $Lemma5\_Proof \triangleq$

$\quad\forall\, s \in Seq(Int):$

$\quad\quad(Len(s) > 0) \Rightarrow$

$\quad\quad\quad SeqSum(s) = SeqSum(Front(s)) + s[Len(s)]$

$\langle 1\rangle$ DEFINE $P(N) \triangleq \forall\, s \in Seq(Int): (Len(s) = N) \Rightarrow (SeqSum(s) = \text{IF } Len(s) = 0 \text{ THEN } 0 \text{ ELSE } SeqSum(Front(s)) + s[Len(s)])$

$\langle 1\rangle 1.\ P(0)$

$\langle 2\rangle$ SUFFICES ASSUME NEW $s \in Seq(Int)$, $Len(s) = 0$
PROVE $SeqSum(s) = \text{IF } Len(s) = 0 \text{ THEN } 0 \text{ ELSE } SeqSum(Front(s)) + s[Len(s)]$

BY $Zenon$ DEF P

$\langle 2\rangle$ QED

BY $s = \langle\rangle$, $Lemma1$, $Zenon$

$\langle 1\rangle 2.$ ASSUME NEW $N \in Nat$, $P(N)$
PROVE $P(N+1)$

$\langle 2\rangle$ SUFFICES ASSUME NEW $s \in Seq(Int)$, $Len(s) = (N+1)$
PROVE $SeqSum(s) = \text{IF } Len(s) = 0 \text{ THEN } 0 \text{ ELSE } SeqSum(Front(s)) + s[Len(s)]$

BY DEF P

$\langle 2\rangle$ SUFFICES $SeqSum(s) = SeqSum(Front(s)) + s[Len(s)]$

OBVIOUS

$\langle 2\rangle 1.\ \wedge Front(s) \in Seq(Int)$
$\wedge Len(Front(s)) = N$

BY $Lemma2$, $N + 1 > 0$, $(N+1) - 1 = N$, $Zenon$

$\langle 2\rangle$ DEFINE $t \triangleq Tail(s)$

$\langle 2\rangle$ USE $FrontDef$

$\langle 2\rangle 2.\ \wedge t \in Seq(Int)$
$\wedge Len(t) = N$
$\wedge SeqSum(s) = s[1] + SeqSum(t)$

BY $HeadTailProperties$, $Lemma1$, $s \neq \langle\rangle$

$\langle 2\rangle 3.$ CASE $N = 0$

$\langle 3\rangle$ USE $\langle 2\rangle 3$

$\langle 3\rangle$ HIDE $FrontDef$ DEF $Front$

$\langle 3\rangle 1.\ SeqSum(Front(s)) = 0$

BY $Lemma1$, $\langle 2\rangle 1$, $Front(s) = \langle\rangle$, $Zenon$

$\langle 3\rangle 2.\ Len(Tail(s)) = 0$

BY $HeadTailProperties$

$\langle 3\rangle 3.\ SeqSum(Tail(s)) =$

$\quad\quad$ IF $Tail(s) = \langle\rangle$ THEN 0 ELSE $Tail(s)[1] + SeqSum(Tail(Tail(s)))$

$\quad$ BY $\langle 2\rangle 2$, $Lemma1$, $Zenon$

$\langle 3\rangle 4.$ $SeqSum(Tail(s)) = 0$

$\quad$ BY $\langle 3\rangle 2$, $\langle 2\rangle 2$, $EmptySeq$, $Tail(s) = \langle\rangle$, $\langle 3\rangle 3$

$\langle 3\rangle 5.$ QED

$\quad$ BY $\langle 2\rangle 2$, $\langle 3\rangle 1$, $\langle 3\rangle 4$

$\langle 2\rangle 4.$CASE $N > 0$

$\langle 3\rangle$ $\land Front(s) \in Seq(Int)$

$\quad\land Front(t) \in Seq(Int)$

$\quad\land Tail(Front(s)) \in Seq(Int)$

$\langle 4\rangle 1.$ $Front(s) \in Seq(Int)$

$\quad$ BY $\langle 2\rangle 4$, $\langle 2\rangle 2$, $Lemma2$

$\langle 4\rangle 2.$ $Front(t) \in Seq(Int)$

$\quad$ BY $\langle 2\rangle 4$, $\langle 2\rangle 2$, $Lemma2$, $Zenon$

$\langle 4\rangle 3.$ $Tail(Front(s)) \in Seq(Int)$

$\quad$ BY $\langle 2\rangle 4$, $Lemma2$

$\langle 4\rangle 4.$ QED

$\quad$ BY $\langle 4\rangle 1$, $\langle 4\rangle 2$, $\langle 4\rangle 3$

$\langle 3\rangle 1.$ $SeqSum(t) = SeqSum(Front(t)) + t[N]$

$\quad$ BY $\langle 1\rangle 2$, $\langle 2\rangle 2$, $\langle 2\rangle 4$, Isa

$\langle 3\rangle 2.$ $SeqSum(t) = SeqSum(Tail(Front(s))) + t[N]$

$\quad$ BY $\langle 3\rangle 1$, $\langle 2\rangle 4$, $Len(s) > 1$, $Lemma3$, $Zenon$

$\langle 3\rangle 3.$ $t[N] = s[N+1]$

$\quad$ BY $\langle 2\rangle 2$, $\langle 2\rangle 4$

$\langle 3\rangle$ HIDE DEF $Front$

$\langle 3\rangle 4.$ $\land SeqSum(s) \in Int$

$\quad\land SeqSum(t) \in Int$

$\quad\land SeqSum(Tail(Front(s))) \in Int$

$\quad\land t[N] \in Int$

$\quad\land s[1] \in Int$

$\langle 4\rangle 1.$ $SeqSum(s) \in Int$

$\quad$ BY $\langle 2\rangle 4$, $\langle 2\rangle 2$, $\langle 2\rangle 1$, $Lemma4$

$\langle 4\rangle 2.$ $SeqSum(t) \in Int$

$\quad$ BY $\langle 2\rangle 4$, $\langle 2\rangle 2$, $\langle 2\rangle 1$, $Lemma4$, $Zenon$

$\langle 4\rangle 3.$ $SeqSum(Tail(Front(s))) \in Int$

$\langle 5\rangle 1.$ $Len(s) > 1$

$\quad$ BY $\langle 2\rangle 4$

$\langle 5\rangle 2.$ $Len(Front(s)) > 0$

$\quad$ BY $\langle 5\rangle 1$, $FrontDef$ DEF $Front$

$\langle 5\rangle 3.$ $Front(s) \neq \langle\rangle$

$\quad$ BY $\langle 5\rangle 2$

$\langle 5\rangle 4.$ $Tail(Front(s)) \in Seq(Int)$

$\quad$ BY $\langle 5\rangle 3$

$\langle 5\rangle 5.$ QED

$\quad$ BY $\langle 2\rangle 4$, $\langle 2\rangle 2$, $\langle 2\rangle 1$, $\langle 5\rangle 3$, $Lemma4$, $Zenon$

$\langle 4 \rangle 4.\ t[N] \in Int$

BY $\langle 2 \rangle 4,\ \langle 2 \rangle 2,\ \langle 2 \rangle 1$

$\langle 4 \rangle 4a.\ s[1] \in Int$

BY $\langle 2 \rangle 4$

$\langle 4 \rangle 5.$ QED

BY $\langle 4 \rangle 1,\ \langle 4 \rangle 2,\ \langle 4 \rangle 3,\ \langle 4 \rangle 4$

$\langle 3 \rangle 5.\ SeqSum(s) = s[1] + SeqSum(Tail(Front(s))) + t[N]$

$\langle 4 \rangle 1.\ SeqSum(s) = s[1] + SeqSum(t)$

BY $\langle 2 \rangle 2$

$\langle 4 \rangle 2.$ QED

BY $\langle 4 \rangle 1,\ \langle 3 \rangle 2,\ \langle 3 \rangle 4,\ Lemma4,\ Z3$

$\langle 3 \rangle 6.\ t[N] = s[N+1]$

BY $\langle 2 \rangle 4$

$\langle 3 \rangle 7.\ s[1] = Front(s)[1]$

BY $\langle 2 \rangle 4$ DEF $Front$

$\langle 3 \rangle 8.\ SeqSum(Front(s)) = Front(s)[1] + SeqSum(Tail(Front(s)))$

BY $\langle 2 \rangle 4,\ Lemma1$

$\langle 3 \rangle 9.$ QED

BY $\langle 3 \rangle 5,\ \langle 3 \rangle 6,\ \langle 3 \rangle 7,\ \langle 3 \rangle 8$

$\langle 2 \rangle 5.$ QED

BY $\langle 2 \rangle 3,\ \langle 2 \rangle 4$

$\langle 1 \rangle 3.\ \forall N \in Nat : P(N)$

BY $\langle 1 \rangle 1,\ \langle 1 \rangle 2,\ NatInduction,\ Isa$

$\langle 1 \rangle 4.$ QED

BY $\langle 1 \rangle 3$

* Modification History

* Created *Fri Apr* 19 14:13:06 *PDT* 2019 by lamport