

1 |----- MODULE *AdditionalSequenceOperators* -----|

3 Copyright: <https://github.com/bringhurst/tlaplus/blob/master/org.lamport.tla.toolbox.uitest/farsite/AdditionalSequenceOperators.tla>

4 |

6 EXTENDS *Naturals, Sequences, FiniteSets*

The TLA+ *Sequences* module defines the operators *Head* and *Tail* for retrieving the first element of a sequence and all-but-the-first elements of a sequence, respectively. This module provides four operators that slightly generalize the notions of *Head* and *Tail*:

First returns the first element of a sequence, equivalently to *Head*. *Last* returns the last element of a sequence. *AllButFirst* returns all-but-the-first elements of a sequence, equivalently to *Tail*.

AllButLast returns all-but-the-last elements of a sequence.

This module also provides several additional operators on sequences: *IsElementInSeq* is a predicate that is true when the specified value is an element of the specified sequence. *IsSequenceOfSetElements* is a predicate that is true when the specified sequence contains all and only elements of the specified set. *IsSortedSequenceOfSetElements* is a predicate that is true when the *IsSequenceOfSetElements* is true and the sequence is also sorted in increasing order. *DeleteElement* produces a sequence by deleting an indicated element from another sequence.

29 Defn $Prepend(s, e) \triangleq \langle e \rangle \circ s$

31 Defn $First(seq) \triangleq seq[1]$

33 Defn $Last(seq) \triangleq seq[Len(seq)]$

35 Defn $AllButFirst(seq) \triangleq [i \in 1 \dots (Len(seq) - 1) \mapsto seq[(i + 1)]]$

37 Defn $AllButLast(seq) \triangleq [i \in 1 \dots (Len(seq) - 1) \mapsto seq[i]]$

39 Defn $DoesSeqPrefixSeq(seq1, seq2) \triangleq$
40 $\wedge Len(seq1) \leq Len(seq2)$
41 $\wedge (\forall i \in 1 \dots Len(seq1) : seq1[i] = seq2[i])$

43 Defn $DoesSeqProperlyPrefixSeq(seq1, seq2) \triangleq$
44 $\wedge Len(seq1) < Len(seq2)$
45 $\wedge (\forall i \in 1 \dots Len(seq1) : seq1[i] = seq2[i])$

47 Defn $IsElementInSeq(el, seq) \triangleq \exists i \in \text{DOMAIN } seq : seq[i] = el$

49 Defn $IsSequenceOfSetElements(seq, set) \triangleq$
50 $\wedge Len(seq) = Cardinality(set)$
51 $\wedge (\forall el \in set : IsElementInSeq(el, seq))$

53 Defn $IsSortedSequenceOfSetElements(seq, set) \triangleq$
54 $\wedge IsSequenceOfSetElements(seq, set)$
55 $\wedge (\forall i \in \text{DOMAIN } seq, j \in \text{DOMAIN } seq : i < j \Rightarrow seq[i] < seq[j])$

57 Defn $DeleteElement(seq, index) \triangleq$
58 $[i \in 1 \dots (Len(seq) - 1) \mapsto \text{IF } i < index \text{ THEN } seq[i] \text{ ELSE } seq[(i + 1)]]$

```

60 Defn IsSorted2Partition( $n, seq1, seq2$ )  $\triangleq$ 
61    $\wedge seq1 \in Seq(1 .. n)$ 
62    $\wedge seq2 \in Seq(1 .. n)$ 
63    $\wedge n = Len(seq1) + Len(seq2)$ 
64    $\wedge (\forall i \in DOMAIN seq1, j \in DOMAIN seq1 : i < j \Rightarrow seq1[i] < seq1[j])$ 
65    $\wedge (\forall i \in DOMAIN seq2, j \in DOMAIN seq2 : i < j \Rightarrow seq2[i] < seq2[j])$ 
66    $\wedge (\forall i \in DOMAIN seq1, j \in DOMAIN seq2 : seq1[i] \neq seq2[j])$ 

68 Defn IsSequenceInterleaving( $seq, subSeq1, subSeq2, indSeq1, indSeq2$ )  $\triangleq$ 
69    $\wedge indSeq1 \in Seq(Nat)$ 
70    $\wedge indSeq2 \in Seq(Nat)$ 
71    $\wedge IsSorted2Partition(Len(seq), indSeq1, indSeq2)$ 
72    $\wedge Len(indSeq1) = Len(subSeq1)$ 
73    $\wedge Len(indSeq2) = Len(subSeq2)$ 
74    $\wedge (\forall i \in DOMAIN indSeq1 : seq[indSeq1[i]] = subSeq1[i])$ 
75    $\wedge (\forall i \in DOMAIN indSeq2 : seq[indSeq2[i]] = subSeq2[i])$ 
76 ]

```

```

\ * Modification History
\ * Last modified Tue Jul 03 15:22:02 CST 2018 by hengxin
\ * Created Tue Jul 03 15:21:02 CST 2018 by hengxin

```