MODULE *RadixTreesValidation*

EXTENDS *FiniteSets*, *Integers*, *RadixTrees*, *Sequences*, *TLC*

Set of characters to use for the alphabet of generated strings.

CONSTANT *Alphabet*

Length of input strings generated

CONSTANT *MinLength*, *MaxLength*

ASSUME

$\land \{MinLength,\ MaxLength\} \subseteq Nat$
$\land\ MinLength \leq MaxLength$
$\land\ MinLength > 0$

Number of unique elements to construct the radix tree with. This
is a set of numbers so you can test with inputs of multiple sizes.

CONSTANT *ElementCounts*

ASSUME $ElementCounts \subseteq Nat$

Inputs is the set of input strings valid for the tree.

$Inputs \triangleq \text{UNION } \{[1\ ..\ n \to Alphabet] : n \in MinLength\ ..\ MaxLength\}$

InputSets is the full set of possible inputs we can send to the radix tree.

$InputSets \triangleq \{T \in \text{SUBSET } Inputs : Cardinality(T) \in ElementCounts\}$

Trees are the set of all radix trees for our inputs. This is a sequence
where $s[1]$ is the input and $s[2]$ is the tree. We keep the input for testing.

$Trees \triangleq \{\langle input,\ RadixTree(input)\rangle : input \in InputSets\}$

All leaf nodes should be values, there is no such thing as a leaf
node that doesn't represent a value.

RECURSIVE $LeafsAreValues(\_)$

$LeafsAreValues(T) \triangleq$

$\lor\ \land Cardinality(\text{DOMAIN } T.Edges) > 0$ if it has edges, its leaves must be values
$\quad\land \forall\, e \in \text{DOMAIN } T.Edges : LeafsAreValues(T.Edges[e])$
$\lor\ \land Cardinality(\text{DOMAIN } T.Edges) = 0$ if it has no edges, it must be a value
$\quad\land Len(T.Value) > 0$

The range of a radix tree should be the set of its inputs.

$RangeIsInput(input,\ tree) \triangleq input = Range(tree)$

The expression that should be checked for validity in the model.

$Valid \triangleq$

$\forall\, pair \in Trees :$
$\quad \text{LET } input \triangleq pair[1] tree \triangleq pair[2] \text{IN}$

$\lor \land RangeIsInput(input,\ tree)$
$\quad \land LeafsAreValues(tree)$
$\lor Print(\langle input,\ tree\rangle,\ \text{FALSE})$

* Modification History
* Last modified *Fri Jul* 02 13:48:34 *PDT* 2021 by *mitchellh*
* Created *Tue Jun* 29 08:02:38 *PDT* 2021 by *mitchellh*