A Unique Representation of Sets

0 Introduction

Some time ago Jan Friso Groote posed the following problem: how to implement (finite) sets, with operations for is-element-of, insertion, and deletion, in such a way that the underlying representation is *unique*? The universe of which these sets are subsets may be assumed to be totally *ordered*, but nevertheless he suspected that the operations for insertion and deletion would not be implementable with a better than linear time complexity.

Given these considerations, a representation by means of increasing arrays is the first solution that comes to mind. In this representation insertion and deletion have linear time complexity indeed, whereas the is-element-of function can be implemented with logarithmic time complexity, namely by means of Binary Search.

This solution, however, has two drawbacks. Firstly, JFG preferred a solution in terms of functional programs; in such a setting the use of (random-access) arrays is awkward. Secondly, although the *worst-case* time complexity of insertion and deletion may not be better than linear, the use of tree-like datastructures may give rise to a better *average-case* time complexity.

Here I present a solution in which, size-balanced and ordered, binary trees are used. As a matter of fact, the choice of datastructure to be used is the largest hurdle: once the decision to use size-balanced and ordered binary trees has been taken, the implementations of the required set operations can be derived in a systematic way, that poses no real difficulties.

In what follows I present these derivations, with a sufficient degree of detail to allow for (human) verification. Readers who are more interested "in de knikkers dan in het spel" are referred to Section 6, where the solutions are recapitulated. The general strategy for the derivations is as follows. For every function I derive a definition satisfying the crucial aspect of its specification – like $[\![\, ins \cdot b \cdot s \,]\!] = \{\, b \,\} \cup [\![\, s \,]\!]$ –; that the function value, which is a tree, also is ordered and size-balanced then remain as additional proof obligations. The strategic freedom during the main derivation is exploited so as to satisfy these additional proof obligations, but I do not include all proofs in this text. Generally, and luckily, these proofs are not difficult.

disclaimer: I do not consider this a scientific study, and I have not investigated to what extent the solutions presented here already exist.

In addition, I do not consider these solutions very interesting from a scientific and/or methodological point of view: they are just the result of applying of standard – at least for me – *design techniques*, although possibly not as well-known as they deserve, and their construction has required little ingenuity. What is interesting, however, is that this demonstrates, once more, the effectiveness of these design techniques and the importance of formulating clear formal specifications.

□

1 Specification

Starting point is a (given) datatype A – the "element type" –, with a (given) total order relation, here simply denoted by "$<$".

To represent the finite subsets of A we need a datatype PA, say, with an abstraction function $[\![\cdot]\!]$, of type $PA \to \mathcal{P}(A)$, such that $[\![s]\!]$ is the subset of A represented by s, for all $s \in PA$.

Somewhat uncoventional is the additional requirement of *uniqueness* of the representation. Formally, this means that the abstraction function $[\![\cdot]\!]$ must be *injective*:

(0) $\quad [\![s]\!] = [\![t]\!] \;\Rightarrow\; s = t$, for all $s, t \in PA$.

The operations to be implemented are the following three functions, with their types:

$$\begin{aligned} &isel : A \to PA \to \mathsf{Bool} \\ &ins : A \to PA \to PA \\ &del : A \to PA \to PA \end{aligned}$$

These functions have the following specifications, for all $b \in A$ and $s \in PA$:

(1) $\quad isel \cdot b \cdot s \;\Leftrightarrow\; b \in [\![s]\!]$

(2) $\quad [\![ins \cdot b \cdot s]\!] = \{ b \} \cup [\![s]\!]$

(3) $\quad [\![del \cdot b \cdot s]\!] = [\![s]\!] \setminus \{ b \}$

2 Representation

2.0 trees

We consider the datatype T of finite binary trees each node of which contains a single value of the element type, A. That is, T is defined recursively, as the smallest set satisfying:

$$\langle\,\rangle \in T \quad \text{, and:}$$
$$\langle s,c,t\rangle \in T \quad \text{, for all } c\in A \text{ and } s,t\in T \quad .$$

Elements in T are called "trees", $\langle\,\rangle$ is called the "empty tree" whereas any tree $\langle s,c,t\rangle$ is called a "composite tree".

Every tree s in T represents a finite subset $[\![\, s \,]\!]$ of A, where the abstraction function $[\![\cdot]\!]$ is defined by, for all $c\in A$ and $s,t\in T$:

$$\begin{array}{lcl} [\![\langle\,\rangle]\!] & = & \phi \\ [\![\langle s,c,t\rangle]\!] & = & [\![\, s \,]\!] \cup \{\, c \,\} \cup [\![\, t \,]\!] \end{array}$$

From this definition it follows immediately that the is-element-of relation $\in$ has the following property:

property (4): for all $b,c\in A$ and $s,t\in T$:

$$\begin{array}{lcl} b\in [\![\langle\,\rangle]\!] & \Leftrightarrow & \mathsf{false} \\ b\in [\![\langle s,c,t\rangle]\!] & \Leftrightarrow & b\in [\![\, s \,]\!] \;\vee\; b=c \;\vee\; b\in [\![\, t \,]\!] \end{array}$$

□

2.1 naming & notational conventions

In what follows variables b,c,d and m have type A and variables r,s,t,u,v have type T. We use $b\in s$ as abbreviation for $b\in [\![\, s \,]\!]$. The minimum and maximum of b,c are denoted by $b{\downarrow}c$ and $b{\uparrow}c$ respectively. The minimum and maximum of an (ordered and finite) set U are denoted by $\mathsf{max}\cdot U$ and $\mathsf{min}\cdot U$ respectively. For example, we now have: $b{\uparrow}c = \mathsf{max}\cdot\{\, b,c \,\}$.

2.2 order and balance

The "ordered" trees are the trees in T that satisfy predicate O, defined by:

$$(5) \qquad O\cdot\langle\,\rangle \;\Leftrightarrow\; \mathsf{true}$$
$$(6) \qquad O\cdot\langle s,c,t\rangle \;\Leftrightarrow\; (\forall b : b\in s : b<c) \;\wedge\; (\forall b : b\in t : c<b) \;\wedge\; O\cdot s \;\wedge\; O\cdot t$$

Ordered trees have the property that all their elements are different. Therefore, we can meaningfully speak of the *number of elements* of an ordered tree. The number of elements in tree s is denoted by $\#s$; a recursive definition is:

$$\begin{array}{lcl} \#\langle\,\rangle & = & 0 \\ \#\langle s,c,t\rangle & = & \#s + 1 + \#t \end{array}$$

property (7): $\#s = \#[\![s]\!]$, for all $s \in T$ satisfying $O{\cdot}s$.
□

The "balanced" trees are the trees in T that satisfy predicate B, defined by:

(8) $B{\cdot}\langle\,\rangle \quad \Leftrightarrow \ \mathsf{true}$

(9) $B{\cdot}\langle s,c,t\rangle \ \Leftrightarrow \ \#t \le \#s \ \wedge \ \#s \le 1+\#t \ \wedge \ B{\cdot}s \ \wedge \ B{\cdot}t$

In what follows we use OB to denote the subset of T of those trees that satisfy both O and B; that is, OB is the subtype of T of those trees that are both ordered and balanced.

Type OB now is the representation type we need, called PA in Section 1. In what follows all variables r, s, t, u, v have type OB.

property (10): Every composite tree $\langle s,c,t\rangle$ in OB satisfies:

$\#s = (\#\langle s,c,t\rangle)\,\mathsf{div}\,2$ and $\#t = (\#\langle s,c,t\rangle - 1)\,\mathsf{div}\,2$.

Hence, the sizes of the subtrees of a composite tree can be calculated from the size of that composite tree.
□

2.3 uniqueness of the representation

We must prove that the representation chosen above satisfies uniqueness requirement (0). To this end we observe that (finite) sets can also be represented uniquely by (finite) increasing lists. That is, without further proof, we postulate that (finite) increasing lists x, y satisfy – here $[\![x]\!]$ denotes the set represented by list x –:

(11) $[\![x]\!] = [\![y]\!] \ \Rightarrow \ x = y$, for all x, y .

In view of this, it suffices to prove that OB-trees represent increasing lists in a unique way. To this end, we define an abstraction function *list*, say, that maps OB-trees to increasing lists, as follows – this just defines the *in-order traversal* of the trees –:

$$\begin{aligned} \mathit{list}{\cdot}\langle\,\rangle &= [\,] \\ \mathit{list}{\cdot}\langle s,c,t\rangle &= \mathit{list}{\cdot}s \mathbin{+\!\!+} [c] \mathbin{+\!\!+} \mathit{list}{\cdot}t \end{aligned}$$

Thus defined, function *list* has the following, rather obvious, properties:

(12) $\#\,(\mathit{list}{\cdot}s) \;=\; \#s$, for all $s{\in}T$.

(13) $[\![\, s \,]\!] = [\![\, \mathit{list}{\cdot}s \,]\!]$, for all $s{\in}T$.

In addition, for ordered trees s we have that $\mathit{list}{\cdot}s$ is increasing. Therefore, as stipulated above, these lists represent sets in a unique way; in terms of OB-trees this can be formulated thus:

(14) $[\![\, \mathit{list}{\cdot}s \,]\!] = [\![\, \mathit{list}{\cdot}t \,]\!] \;\Rightarrow\; \mathit{list}{\cdot}s \;=\; \mathit{list}{\cdot}t$, for all $s,t{\in}\mathit{OB}$.

Finally, we need a crucial property of finite lists, namely that if two lists are equal then all their finite prefixes (and suffixes) *of the same length* are (pair-wise) equal too:

(15) $(\mathit{x0} +\!\!+\, \mathit{y0} = \mathit{x1} +\!\!+\, \mathit{y1}) \;\wedge\; (\#\mathit{x0} = \#\mathit{x1}) \;\Rightarrow\; (\mathit{x0} = \mathit{x1}) \;\wedge\; (\mathit{y0} = \mathit{y1})$,
for all finite lists $\mathit{x0}, \mathit{y0}, \mathit{x1}, \mathit{y1}$.

Now we have all ingredients needed to prove requirement (0). This proof obligation consists of two, disjoint, parts: one for the empty set and one for the nonempty sets. The representation of the empty set obviously is unique, because there is only one empty tree, and all composite trees represent non-empty sets. The proof is by Structural Induction, so we use the property to be proved for the (composite) tree's subtrees. For composite OB-trees we derive:

$[\![\, \langle s,c,t\rangle \,]\!] = [\![\, \langle u,d,v\rangle \,]\!]$
$\Leftrightarrow$ { property (13) }
$[\![\, \mathit{list}{\cdot}\langle s,c,t\rangle \,]\!] = [\![\, \mathit{list}{\cdot}\langle u,d,v\rangle \,]\!]$
$\Rightarrow$ { property (14), using that $\langle s,c,t\rangle$ and $\langle u,d,v\rangle$ are ordered }
$\mathit{list}{\cdot}\langle s,c,t\rangle \;=\; \mathit{list}{\cdot}\langle u,d,v\rangle$
$\Rightarrow$ { definition of *list* (twice) }
$\mathit{list}{\cdot}s +\!\!+\, [c] +\!\!+\, \mathit{list}{\cdot}t \;=\; \mathit{list}{\cdot}u +\!\!+\, [d] +\!\!+\, \mathit{list}{\cdot}v$
$\Rightarrow$ { property (15) (twice), explanation follows }
$(\mathit{list}{\cdot}s = \mathit{list}{\cdot}u) \;\wedge\; ([c] = [d]) \;\wedge\; (\mathit{list}{\cdot}t = \mathit{list}{\cdot}v)$
$\Rightarrow$ { Induction Hypothesis (twice), and equality of singleton lists }
$(s = u) \;\wedge\; (c = d) \;\wedge\; (t = v)$
$\Rightarrow$ { Leibniz }
$\langle s,c,t\rangle \;=\; \langle u,d,v\rangle$.

Property (15) is applicable here: if $[\![\langle s, c, t \rangle]\!] = [\![\langle u, d, v \rangle]\!]$ then we also have $\#[\![\langle s, c, t \rangle]\!] = \#[\![\langle u, d, v \rangle]\!]$, which, on account of property (7), is equivalent to $\#\langle s, c, t \rangle = \#\langle u, d, v \rangle$. By property (10), this implies $\#s = \#u$ and $\#t = \#v$, and, hence, also $\#(list \cdot s) = \#(list \cdot u)$ and $\#(list \cdot t) = \#(list \cdot v)$, as required.

3 Is-element-of

Function *isel* is a standard one in the realm of search trees. Its definition follows directly from property 4, by case analysis (for the sake of efficiency): If $b < c$ then $\neg(b = c)$ and $\neg(b \in t)$; so, in this case $b \in \langle s, c, t \rangle$ is equivalent to $b \in s$; and so on. Thus, we obtain the following recursive definition for *isel*:

$$
\begin{array}{lcl}
isel \cdot b \cdot \langle\,\rangle & = & \mathsf{false} \\
isel \cdot b \cdot \langle s, c, t \rangle & = & \mathsf{if}\ \ b < c \ \rightarrow\ isel \cdot b \cdot s \\
 & & []\ \ b = c \ \rightarrow\ \mathsf{true} \\
 & & []\ \ c < b \ \rightarrow\ isel \cdot b \cdot t \\
 & & \mathsf{fi}
\end{array}
$$

4 insertion

4.0 ins

We recall specification (2) of function *ins*, for all b, s:

$$[\![ins \cdot b \cdot s]\!] \ = \ \{b\} \cup [\![s]\!] \ .$$

If $b \in s$ then $\{b\} \cup [\![s]\!]$ equals $[\![s]\!]$; hence, in this special case $ins \cdot b \cdot s = s$ satisfies *ins* 's specification. It so happens that the derivation becomes simpler – less case analysis –, if we treat the case $\neg(b \in s)$ separately, by strengthening the function's precondition. That is, we introduce a new function *inz*, say, with this specification, for all b, s:

$$\neg(b \in s) \ \Rightarrow \ [\![inz \cdot b \cdot s]\!] = \{b\} \cup [\![s]\!] \ .$$

In terms of *inz* function *ins* can now be defined as follows:

$$
\begin{array}{lcl}
ins \cdot b \cdot s & = & \mathsf{if}\ \ \ isel \cdot b \cdot s \ \rightarrow\ s \\
 & & []\ \neg isel \cdot b \cdot s \ \rightarrow\ inz \cdot b \cdot s \\
 & & \mathsf{fi}
\end{array}
$$

4.1 inz

We recall function inz's specification, for all b, s:

$$\neg(b \in s) \;\Rightarrow\; [\![\, inz \cdot b \cdot s \,]\!] = \{b\} \cup [\![\, s \,]\!] \quad .$$

Notice that a definition for the case $inz \cdot b \cdot \langle s, c, t \rangle$ needs no alternative for the case $b = c$, because the precondition, that is $\neg(b \in \langle s, c, t \rangle)$, implies $b \neq c$. A tentative definition for inz could be:

$$\begin{array}{lcl} inz \cdot b \cdot \langle\,\rangle & = & \langle \langle\,\rangle, b, \langle\,\rangle \rangle \\ inz \cdot b \cdot \langle s, c, t \rangle & = & \mathsf{if}\;\; b < c \;\rightarrow\; \langle\, inz \cdot b \cdot s \,, c \,, t \,\rangle \\ & & []\;\; c < b \;\rightarrow\; \langle\, s \,, c \,, inz \cdot b \cdot t \,\rangle \\ & & \mathsf{fi} \end{array}$$

This maintains orderedness of the trees, but not necessarily their balance. We investigate this further for each of the two alternatives separately.

Because $\neg(b \in s)$ we have that $\#\, inz \cdot b \cdot s$ equals $1 + \#s$; so, $\langle\, inz \cdot b \cdot s \,, c \,, t \,\rangle$ is balanced if and only if $\#s = \#t$: in that case we have that $\#\, inz \cdot b \cdot s$ equals $1 + \#t$, which is permitted. The only other possibility is $\#s = 1 + \#t$; in this case $\#\, inz \cdot b \cdot s$ equals $2 + \#t$, which is one too large. Therefore, for this case we derive:

$$\begin{array}{ll} & [\![\, inz \cdot b \cdot \langle s, c, t \rangle \,]\!] \\ = & \quad \{\ \text{specification of } inz\text{, and definition of } [\![\cdot]\!]\ \} \\ & \{b\} \cup [\![\, s \,]\!] \cup \{c\} \cup [\![\, t \,]\!] \\ = & \quad \{\ \text{let } m = \mathsf{max} \cdot (\{b\} \cup [\![\, s \,]\!])\text{: set calculus}\ \} \\ & ((\{b\} \cup [\![\, s \,]\!]) \setminus \{m\}) \cup \{m\} \cup \{c\} \cup [\![\, t \,]\!] \\ = & \quad \{\ \text{let } r \text{ satisfy: } [\![\, r \,]\!] = (\{b\} \cup [\![\, s \,]\!]) \setminus \{m\}\ \} \\ & [\![\, r \,]\!] \cup \{m\} \cup \{c\} \cup [\![\, t \,]\!] \\ = & \quad \{\ \text{specification of } inz\ \} \\ & [\![\, r \,]\!] \cup \{m\} \cup [\![\, inz \cdot c \cdot t \,]\!] \\ = & \quad \{\ \text{definition of } [\![\cdot]\!]\ \} \\ & [\![\, \langle r \,, m \,, inz \cdot c \cdot t \,\rangle \,]\!] \quad , \end{array}$$

so, function value $inz \cdot b \cdot \langle s, c, t \rangle$ can be defined as $\langle\, r \,, m \,, inz \cdot c \cdot t \,\rangle$, provided we add proper definitions for the additional variables r and m.

With respect to r, we observe that the expression $(\{b\} \cup [\![s]\!]) \setminus \{m\}$ can be implemented, in terms of our functions *ins* and *del*, as $del \cdot m \cdot (ins \cdot b \cdot s)$. In addition, for m we might introduce a new function mx, say, satisfying $mx \cdot b \cdot s = \mathsf{max} \cdot (\{b\} \cup [\![s]\!])$.

The use of (at least) three independent function applications on the same subtree has, however, a disastrous effect on the efficiency of the program. The efficiency of the program will be much better if we succeed in combining these function applications into a single one. This should be possible because all functions involved are applied, more or less, to the very same subtree s.

For this purpose, we introduce a function *insdelmx*, say, such that the value of $insdelmx \cdot b \cdot s$ is a pair consisting of the tree representing the set $(\{b\} \cup [\![s]\!]) \setminus \{m\}$ together with value m, where $m = \mathsf{max} \cdot (\{b\} \cup [\![s]\!])$. In addition, by symmetry, we also need a function *insdelmn*, say, such that the value of $insdelmn \cdot b \cdot s$ is a pair consisting of the tree representing the set $(\{b\} \cup [\![s]\!]) \setminus \{m\}$ and m, where $m = \mathsf{min} \cdot (\{b\} \cup [\![s]\!])$.

By the above considerations and using the new functions we can now define function *inz* in the following way:

$$
\begin{array}{lcl}
inz \cdot b \cdot \langle\,\rangle & = & \langle \langle\,\rangle, b, \langle\,\rangle \rangle \\
inz \cdot b \cdot \langle s, c, t \rangle & = & \mathsf{if}\ \ b < c \wedge \#s = \#t \quad \rightarrow\ \ \langle\, inz \cdot b \cdot s\,,\, c\,,\, t\,\rangle \\
 & & []\ \ b < c \wedge \#s = 1 + \#t\ \rightarrow \\
 & & \quad \langle\, r\,,\, m\,,\, inz \cdot c \cdot t\,\rangle\ \ \mathsf{whr}\ \langle r, m\rangle = insdelmx \cdot b \cdot s\ \mathsf{end} \\
 & & []\ \ c < b \wedge \#s = \#t \quad \rightarrow \\
 & & \quad \langle\, inz \cdot c \cdot s\,,\, m\,,\, r\,\rangle\ \ \mathsf{whr}\ \langle r, m\rangle = insdelmn \cdot b \cdot t\ \mathsf{end} \\
 & & []\ \ c < b \wedge \#s = 1 + \#t\ \rightarrow\ \ \langle\, s\,,\, c\,,\, inz \cdot b \cdot t\,\rangle \\
 & & \mathsf{fi}
\end{array}
$$

remark on implementation: The relation between $\#s$ and $\#t$ is very restricted: either $\#s = \#t$ or $\#s = 1 + \#t$. Hence, the expression $\#s - \#t$ will only have two values: either 0 or 1. Therefore, to evaluate the guards in the above definition it is not necessary to *compute* $\#s$ and #t: it suffices to extend the tuples representing the trees with the additional value $\#s - \#t$; this requires only a single additional bit per node of the trees.

Alternatively, if the size of the whole tree is available in the context where function *inz* is applied, this function can also be equipped with an additional parameter representing this size. Property (10) can then be used to calculate the sizes of the subtrees, to be used in the guards and to be passed down in the recursive applications of *inz*.

□

* * *

We have introduced a function *insdelmx* , with type $A \to OB \to \langle\, OB, A \rangle$, and from its above use we obtain this specification, for all b, s, r, m :

$$\neg(b \in s) \;\wedge\; \mathit{insdelmx} \cdot b \cdot s \;=\; \langle r, m \rangle \;\Rightarrow$$
$$[\![\, r \,]\!] = (\,\{\, b \,\} \cup [\![\, s \,]\!]\,) \setminus \{\, m \,\} \;\wedge\; m = \mathsf{max} \cdot (\,\{\, b \,\} \cup [\![\, s \,]\!]\,)$$

Notice that this specification implies that $\#r \;=\; \#s$, because of the precondition $\neg(b \in s)$. So, the value of function *insdelmx* contains a tree of the same size as its argument.

Similarly, function *insdelmn* has the same type and has this specification, for all b, s, r, m :

$$\neg(b \in s) \;\wedge\; \mathit{insdelmn} \cdot b \cdot s \;=\; \langle r, m \rangle \;\Rightarrow$$
$$[\![\, r \,]\!] = (\,\{\, b \,\} \cup [\![\, s \,]\!]\,) \setminus \{\, m \,\} \;\wedge\; m = \mathsf{min} \cdot (\,\{\, b \,\} \cup [\![\, s \,]\!]\,)$$

4.2 insdelmx

By its specification, the value of $\mathit{insdelmx} \cdot b \cdot s$ is a pair $\langle r, m \rangle$ satisfying:

$$m \;=\; \mathsf{max} \cdot (\,\{\, b \,\} \cup [\![\, s \,]\!]\,) \quad \text{, and:}$$

$$[\![\, r \,]\!] \;=\; (\,\{\, b \,\} \cup [\![\, s \,]\!]\,) \setminus \{\, m \,\} \quad .$$

For the case $s := \langle\, \rangle$ we derive, for m and r separately:

$$\begin{array}{ll} & m \\ = & \quad \{\ \text{specification of } m\ \} \\ & \mathsf{max} \cdot (\,\{\, b \,\} \cup [\![\, \langle\, \rangle \,]\!]\,) \\ = & \quad \{\ \text{definition of } [\![\, \cdot \,]\!] \text{; set calculus}\ \} \\ & \mathsf{max} \cdot \{\, b \,\} \\ = & \quad \{\ \text{definition of } \mathsf{max}\ \} \\ & b \quad , \end{array}$$

and:

$$\begin{array}{ll} & [\![\, r \,]\!] \\ = & \quad \{\ \text{specification of } r\ \} \end{array}$$

$$
\begin{array}{ll}
 & (\{b\} \cup [\![\langle\,\rangle]\!]) \setminus \{m\} \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \text{; set calculus } \} \\
 & \{b\} \setminus \{m\} \\
= & \quad \{ \ m = b \text{; set calculus } \} \\
 & \phi \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \ \} \\
 & [\![\langle\,\rangle]\!] \quad .
\end{array}
$$

Thus, we conclude that $\langle \langle\,\rangle, b \rangle$ is a good choice for $insdelmx \cdot b \cdot \langle\,\rangle$.

For the case $s := \langle s, c, t \rangle$ we derive, again for m and r separately:

$$
\begin{array}{ll}
 & m \\
= & \quad \{ \text{ specification of } m \ \} \\
 & \mathsf{max} \cdot (\{b\} \cup [\![\langle s, c, t \rangle]\!]) \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \ \} \\
 & \mathsf{max} \cdot (\{b\} \cup [\![s]\!] \cup \{c\} \cup [\![t]\!]) \\
= & \quad \{ \ \mathsf{max} \text{ over } \cup \ \} \\
 & b \uparrow \mathsf{max} \cdot [\![s]\!] \uparrow c \uparrow \mathsf{max} \cdot [\![t]\!] \\
= & \quad \{ \ \langle s, c, t \rangle \text{ is ordered: } \mathsf{max} \cdot [\![s]\!] < c \ \} \\
 & b \uparrow c \uparrow \mathsf{max} \cdot [\![t]\!] \\
= & \quad \{ \ \mathsf{max} \text{ over } \cup \ \} \\
 & \mathsf{max} \cdot (\{b \uparrow c\} \cup [\![t]\!]) \quad ,
\end{array}
$$

and:

$$
\begin{array}{ll}
 & [\![r]\!] \\
= & \quad \{ \text{ specification of } r \ \} \\
 & (\{b\} \cup [\![\langle s, c, t \rangle]\!]) \setminus \{m\} \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \text{; set calculus } \} \\
 & (\{b\} \cup [\![s]\!] \cup \{c\} \cup [\![t]\!]) \setminus \{m\} \\
= & \quad \{ \ \{b, c\} = \{ b \downarrow c, b \uparrow c \} \ \} \\
 & (\{b \downarrow c\} \cup [\![s]\!] \cup \{b \uparrow c\} \cup [\![t]\!]) \setminus \{m\} \\
= & \quad \{ \ m \in \{b \uparrow c\} \cup [\![t]\!] \text{, as derived above: set calculus } \}
\end{array}
$$

$$
\begin{array}{ll}
 & (\{b{\downarrow}c\} \cup [\![s]\!]) \cup ((\{b{\uparrow}c\} \cup [\![t]\!]) \setminus \{m\}) \\
= & \quad \{ \text{ introduce } c1 \text{ with } c1 = \mathsf{max}\cdot(\{b{\downarrow}c\} \cup [\![s]\!]) \ \} \\
 & ((\{b{\downarrow}c\} \cup [\![s]\!]) \setminus \{c1\}) \cup \{c1\} \cup ((\{b{\uparrow}c\} \cup [\![t]\!]) \setminus \{m\}) \\
= & \quad \{ \text{ introduce } s1 \text{ and } t1, \text{ see below } \} \\
 & [\![s1]\!] \cup \{c1\} \cup [\![t1]\!] \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \ \} \\
 & [\![\langle s1, c1, t1 \rangle]\!] \ ,
\end{array}
$$

from which we conclude that $r = \langle s1, c1, t1 \rangle$ is a good choice.

In this derivation we have assumed that $s1$ and $c1$ satisfy: $c1 = \mathsf{max}\cdot(\{b{\downarrow}c\} \cup [\![s]\!])$ and $s1 = (\{b{\downarrow}c\} \cup [\![s]\!]) \setminus \{c1\}$; these assumptions are met if we define $s1$ and $c1$ by $\langle s1, c1 \rangle = insdelmx\cdot(b{\downarrow}c)\cdot s$.

Moreover, we also have assumed $m = \mathsf{max}\cdot(\{b{\uparrow}c\} \cup [\![t]\!])$ and $t1 = (\{b{\uparrow}c\} \cup [\![t]\!]) \setminus \{m\}$; these assumptions are met if we define $t1$ and m by $\langle t1, m \rangle = insdelmx\cdot(b{\uparrow}c)\cdot t$.

* * *

From the above derivations we obtain the following recursive definition for function *insdelmx*:

$$
\begin{array}{lcl}
insdelmx\cdot b\cdot\langle\,\rangle & = & \langle \langle\,\rangle , b \rangle \\
insdelmx\cdot b\cdot\langle s, c, t \rangle & = & \langle \langle s1, c1, t1 \rangle , m \rangle \\
 & & \mathsf{whr}\ \langle s1, c1 \rangle = insdelmx\cdot(b{\downarrow}c)\cdot s \\
 & & \quad \& \ \langle t1, m \rangle = insdelmx\cdot(b{\uparrow}c)\cdot t \\
 & & \mathsf{end}
\end{array}
$$

Because this definition contains a recursive application of the function to both subtrees, s and t, of the argument $\langle s, c, t \rangle$, the time complexity of this function is *linear* in the size of the tree: during evaluation of this function all nodes of the tree are visited exactly once.

4.3 insdelmn

Function *insdelmn* is the symmetric counterpart of *insdelmx*, under swapping "left" and "right" in the trees and swapping "smaller" and "larger" – and, hence, swapping $\downarrow$ and $\uparrow$ as well – in the values. Thus, a definition for *insdelmn* is obtained from *insdelmx*'s definition without much effort:

$$
\begin{array}{lcl}
insdelmn\cdot b\cdot\langle\,\rangle & = & \langle\langle\,\rangle, b\rangle \\
insdelmn\cdot b\cdot\langle s,c,t\rangle & = & \langle\langle s1, c1, t1\rangle,\, m\,\rangle \\
 & & \quad \mathsf{whr}\ \langle t1, c1\rangle = insdelmn\cdot(b{\uparrow}c)\cdot t \\
 & & \quad\ \ \&\ \ \langle s1, m\rangle = insdelmn\cdot(b{\downarrow}c)\cdot s \\
 & & \quad \mathsf{end}
\end{array}
$$

5 Deletion

5.0 del

We recall specification (3) for function *del*, for all b, s:

$$[\![\, del\cdot b\cdot s \,]\!] = [\![\, s \,]\!] \setminus \{\, b \,\} \quad .$$

As was the case with function *ins*, it is convenient to distinguish the cases $b \in s$ and $\neg(b \in s)$, and to strengthen the function's precondition with $b \in s$. This gives rise to a new function *dlz*, say, with this specification, for all b, s:

$$b \in s \ \Rightarrow\ [\![\, dlz\cdot b\cdot s \,]\!] = [\![\, s \,]\!] \setminus \{\, b \,\} \quad .$$

In terms of *dlz* function *del* now can be defined by:

$$
\begin{array}{lcl}
del\cdot b\cdot s & = & \mathsf{if}\ \neg\, isel\cdot b\cdot s \ \rightarrow\ s \\
 & & [\!]\quad isel\cdot b\cdot s \ \rightarrow\ dlz\cdot b\cdot s \\
 & & \mathsf{fi}
\end{array}
$$

5.1 dlz

In *dlz*'s specification, the precondition $b \in s$ for $dlz\cdot b\cdot s$ implies $s \neq \langle\,\rangle$; hence, we only need a definition for $dlz\cdot b\cdot s$ for composite s, that is, for s of the shape $\langle s, c, t\rangle$. To develop a solution we now must distinguish three cases: $b = c$, $b < c$, and $c < b$.

* * *

The simplest case is $b = c$; then, we have:

$$
\begin{array}{ll}
 & [\![\, \langle s, c, t\rangle \,]\!] \setminus \{\, b \,\} \\
= & \quad \{\ \text{definition of } [\![\,\cdot\,]\!]\ \}
\end{array}
$$

$$
\begin{array}{ll}
 & (\llbracket\, s \,\rrbracket \cup \{\, c \,\} \cup \llbracket\, t \,\rrbracket) \setminus \{\, b \,\} \\
= & \quad \{\ \setminus \text{ over } \cup;\ b = c \text{ and, hence, } \neg(c \in s) \text{ and } \neg(c \in t)\ \} \\
 & \llbracket\, s \,\rrbracket \cup \llbracket\, t \,\rrbracket \ .
\end{array}
$$

To be able to represent the set $\llbracket\, s \,\rrbracket \cup \llbracket\, t \,\rrbracket$ as a tree , some further case analysis is unavoidable. Firstly, the set $\llbracket\, s \,\rrbracket \cup \llbracket\, t \,\rrbracket$ may be empty, namely if both s and t are empty. In this case the expression can only be rewritten to $\llbracket\, \langle\, \rangle \,\rrbracket$. Notice that, because $\langle s, c, t \rangle$ is balanced we have $s = \langle\, \rangle \Rightarrow t = \langle\, \rangle$, so we only need $s = \langle\, \rangle$ as guard for this case. Secondly, if the set $\llbracket\, s \,\rrbracket \cup \llbracket\, t \,\rrbracket$ is non-empty it can only be represented by some composite tree $\langle\, s1, m, t1 \,\rangle$, say, satisfying:

$$
\llbracket\, s \,\rrbracket \cup \llbracket\, t \,\rrbracket \ = \ \llbracket\, s1 \,\rrbracket \cup \{\, m \,\} \cup \llbracket\, t1 \,\rrbracket \ .
$$

Here we may choose either $m \in s$ or $m \in t$, and, subsequently, $s1$ and $t1$ must satisfy either:

$$
\llbracket\, s1 \,\rrbracket = \llbracket\, s \,\rrbracket \setminus \{\, m \,\} \ \wedge \ \llbracket\, t1 \,\rrbracket = \llbracket\, t \,\rrbracket \ , \text{ or:}
$$

$$
\llbracket\, s1 \,\rrbracket = \llbracket\, s \,\rrbracket \ \wedge \ \llbracket\, t1 \,\rrbracket = \llbracket\, t \,\rrbracket \setminus \{\, m \,\} \ , \text{ respectively } .
$$

We let the choice between these two possibilities depend on the relative sizes of subtrees s and t, so as to guarantee that the resulting tree is balanced. In addition, $m \in s$ requires $s \neq \langle\, \rangle$, of course, and $m \in t$ requires $t \neq \langle\, \rangle$; this must be incorporated in the guards of these alternatives.

We also must see to it that the resulting tree is ordered. This is achieved by strengthening $m \in s$ to $m = \mathsf{max} \cdot \llbracket\, s \,\rrbracket$, and by strengthening $m \in t$ to $m = \mathsf{min} \cdot \llbracket\, t \,\rrbracket$.

Thus, we obtain the following solution for the case $b = c$:

$$
\begin{array}{l}
\mathsf{if} \ \ s = \langle\, \rangle \qquad\qquad\quad \rightarrow \ \ \langle\, \rangle \\
[\,] \ \ \#s = \#t \wedge t \neq \langle\, \rangle \ \rightarrow \ \ \langle\, s \,,\, m \,,\, t1 \,\rangle \\
\quad \mathsf{whr} \ \ t1 : \llbracket\, t1 \,\rrbracket = \llbracket\, t \,\rrbracket \setminus \{\, m \,\} \ \ \&\ \ m = \mathsf{min} \cdot \llbracket\, t \,\rrbracket \ \ \mathsf{end} \\
[\,] \ \ \#s = 1 + \#t \qquad\quad \rightarrow \ \ \langle\, s1 \,,\, m \,,\, t \,\rangle \\
\quad \mathsf{whr} \ \ s1 : \llbracket\, s1 \,\rrbracket = \llbracket\, s \,\rrbracket \setminus \{\, m \,\} \ \ \&\ \ m = \mathsf{max} \cdot \llbracket\, t \,\rrbracket \ \ \mathsf{end} \\
\mathsf{fi}
\end{array}
$$

To implement the definitions of the local variables $s1$, $t1$, and m , we combine – again: for the sake of efficiency – expressions $\llbracket\, t \,\rrbracket \setminus \{\, m \,\}$ and $\mathsf{min} \cdot \llbracket\, t \,\rrbracket$ into the application of a single function *delmn* , say, with this specification, for all b, t, r, m:

$$t \neq \langle\,\rangle \;\wedge\; delmn \cdot t = \langle r, m\rangle \;\Rightarrow\; [\![\, r \,]\!] = [\![\, t \,]\!] \setminus \{\, m \,\} \;\wedge\; m = \mathsf{min} \cdot [\![\, t \,]\!] \quad .$$

Similarly, to combine $[\![\, s \,]\!] \setminus \{\, m \,\}$ and $\mathsf{max} \cdot [\![\, s \,]\!]$ we introduce a function $delmx$, with this specification, for all b, s, r, m:

$$s \neq \langle\,\rangle \;\wedge\; delmx \cdot s = \langle r, m\rangle \;\Rightarrow\; [\![\, r \,]\!] = [\![\, s \,]\!] \setminus \{\, m \,\} \;\wedge\; m = \mathsf{max} \cdot [\![\, s \,]\!] \quad .$$

In terms of these new functions the above solution can be implemented thus:

$$\begin{array}{lcl}
\mathsf{if}\;\; s = \langle\,\rangle & \rightarrow & \langle\,\rangle \\
[]\;\; \#s = \#t \wedge t \neq \langle\,\rangle & \rightarrow & \langle\, s\,,m\,,t1\,\rangle \;\; \mathsf{whr}\; \langle t1, m\rangle = delmn \cdot t\; \mathsf{end} \\
[]\;\; \#s = 1 + \#t & \rightarrow & \langle\, s1\,,m\,,t\,\rangle \;\; \mathsf{whr}\; \langle s1, m\rangle = delmx \cdot s\; \mathsf{end} \\
\mathsf{fi}
\end{array}$$

* * *

The next main case is $b < c$. If $\#s = 1 + \#t$ then the obvious solution, $\langle\, dlz \cdot b \cdot s\,, c\,, t\,\rangle$, satisfies all requirements. If $\#s = \#t$ some rebalancing is needed. For this purpose we introduce a function $mxinsdel$, with this specification, for all m, b, s:

$$\begin{array}{l}
(\forall c : c \in s : c < m) \;\wedge\; b \in s \;\Rightarrow \\
[\![\; mxinsdel \cdot m \cdot b \cdot s \;]\!] \;\; = \{\, m \,\} \cup ([\![\, s \,]\!] \setminus \{\, b \,\})
\end{array}$$

In terms of this function the expression $\langle\, mxinsdel \cdot c \cdot b \cdot s\,, m\,, t1\,\rangle$ represents $[\![\, \langle s, c, t\rangle \,]\!] \setminus \{\, b \,\}$, and is both ordered and balanced as required, provided m and $t1$ are defined by $\langle t1, m\rangle = delmn \cdot t$.

* * *

The third (and last) main case is $c < b$. This is the symmetric counterpart of the previous case, and as such it poses no new problems. It requires the introduction of the symmetric counterpart of function $mxinsdel$, which we call $mninsdel$ and which is specified by, for all m, b, t:

$$\begin{array}{l}
(\forall c : c \in t : m < c) \;\wedge\; b \in t \;\Rightarrow \\
[\![\; mninsdel \cdot m \cdot b \cdot t \;]\!] \;\; = \{\, m \,\} \cup ([\![\, t \,]\!] \setminus \{\, b \,\})
\end{array}$$

Now, if $\#s = \#t$ the obvious proposal $\langle\, s\,, c\,, dlz \cdot b \cdot t\,\rangle$ does the job, and if $\#s = 1 + \#t$ then $\langle\, s1\,, m\,, mninsdel \cdot c \cdot b \cdot t\,\rangle$ meets all requirements, where $\langle s1, m\rangle = delmx \cdot s$.

* * *

By collecting all results and combining them into a single definition we obtain the following definition for function dlz :

$$
\begin{array}{l}
dlz{\cdot}b{\cdot}\langle s,c,t\rangle \;=\\
\mathsf{if}\;\; b<c \wedge \#s=1+\#t \;\rightarrow\; \langle\, dlz{\cdot}b{\cdot}s\,,c\,,t\,\rangle\\
[]\;\; b<c \wedge \#s=\#t \;\rightarrow\; \langle\, mxinsdel{\cdot}c{\cdot}b{\cdot}s\,,m\,,t1\,\rangle\\
\qquad\qquad\qquad\qquad\quad \mathsf{whr}\;\langle t1,m\rangle = delmn{\cdot}t\;\mathsf{end}\\
[]\;\; b=c \wedge s=\langle\,\rangle \;\rightarrow\; \langle\,\rangle\\
[]\;\; b=c \wedge \#s=\#t \wedge t\neq\langle\,\rangle \;\rightarrow\; \langle\, s\,,m\,,t1\,\rangle\\
\qquad\qquad\qquad\qquad\qquad\qquad\quad \mathsf{whr}\;\langle t1,m\rangle = delmn{\cdot}t\;\mathsf{end}\\
[]\;\; b=c \wedge \#s=1+\#t \;\rightarrow\; \langle\, s1\,,m\,,t\,\rangle\\
\qquad\qquad\qquad\qquad\quad \mathsf{whr}\;\langle s1,m\rangle = delmx{\cdot}s\;\mathsf{end}\\
[]\;\; c<b \wedge \#s=\#t \;\rightarrow\; \langle\, s\,,c\,,dlz{\cdot}b{\cdot}t\,\rangle\\
[]\;\; c<b \wedge \#s=1+\#t \;\rightarrow\; \langle\, s1\,,m\,,mninsdel{\cdot}c{\cdot}b{\cdot}t\,\rangle\\
\qquad\qquad\qquad\qquad\quad \mathsf{whr}\;\langle s1,m\rangle = delmx{\cdot}s\;\mathsf{end}\\
\mathsf{fi}
\end{array}
$$

5.2 mxinsdel

We recall the specification of function $mxinsdel$, for all m,b,s :

$$
\begin{array}{l}
(\forall c:c\in s:c<m)\;\wedge\; b\in s \;\Rightarrow\\
[\![\, mxinsdel{\cdot}m{\cdot}b{\cdot}s\,]\!] \;=\{m\}\cup([\![\, s\,]\!]\setminus\{b\})
\end{array}
$$

The precondition, $b\in s$, of $mxinsdel{\cdot}m{\cdot}b{\cdot}s$ implies that s is non-empty, so we only need a definition for $mxinsdel{\cdot}m{\cdot}b{\cdot}s$ for non-empty s , that is, for trees of the shape $\langle s,c,t\rangle$. The derivation of such a definition requires distinction of the three cases for the relative order of b and c .

* * *

For the case $b<c$ we derive:

$$
\begin{array}{ll}
 & [\![\, mxinsdel{\cdot}m{\cdot}b{\cdot}\langle s,c,t\rangle\,]\!]\\
= & \quad \{\text{ specification of } mxinsdel\ \}
\end{array}
$$

$$
\begin{array}{ll}
 & \{m\} \cup (\llbracket\, \langle s, c, t\rangle \,\rrbracket \setminus \{b\}) \\
= & \quad \{ \text{ definition of } \llbracket\cdot\rrbracket \ \} \\
 & \{m\} \cup ((\llbracket s \rrbracket \cup \{c\} \cup \llbracket t \rrbracket) \setminus \{b\}) \\
= & \quad \{\ b<c \text{, so } b\in s \text{ and } \neg(b\in t) : \text{ set calculus } \} \\
 & \{m\} \cup (\llbracket s \rrbracket \setminus \{b\}) \cup \{c\} \cup \llbracket t \rrbracket \\
= & \quad \{\ c<m : \text{ more set calculus } \} \\
 & \{c\} \cup (\llbracket s \rrbracket \setminus \{b\}) \cup \{m\} \cup \llbracket t \rrbracket \\
= & \quad \{\ (\forall d : d\in s : d<c) \wedge b\in s : \text{ specification of } \mathit{mxinsdel} \ \} \\
 & \llbracket\, \mathit{mxinsdel}\cdot c\cdot b\cdot s \,\rrbracket \cup \{m\} \cup \llbracket t \rrbracket \\
= & \quad \{ \text{ introduce } \mathit{c1} \text{ with } \mathit{c1} = \mathsf{min}\cdot(\{m\} \cup \llbracket t \rrbracket) \ \} \\
 & \llbracket\, \mathit{mxinsdel}\cdot c\cdot b\cdot s \,\rrbracket \cup \{\mathit{c1}\} \cup (\{m\} \cup \llbracket t \rrbracket) \setminus \{\mathit{c1}\} \\
= & \quad \{ \text{ introduce } \mathit{t1} \text{ with } \llbracket\, \mathit{t1} \,\rrbracket = (\{m\} \cup \llbracket t \rrbracket) \setminus \{\mathit{c1}\} \ \} \\
 & \llbracket\, \mathit{mxinsdel}\cdot c\cdot b\cdot s \,\rrbracket \cup \{\mathit{c1}\} \cup \llbracket\, \mathit{t1} \,\rrbracket \\
= & \quad \{ \text{ definition of } \llbracket\cdot\rrbracket \ \} \\
 & \llbracket\, \langle\, \mathit{mxinsdel}\cdot c\cdot b\cdot s \,,\, \mathit{c1} \,,\, \mathit{t1} \,\rangle \,\rrbracket \quad .
\end{array}
$$

Thus, we conclude that $\mathit{mxinsdel}\cdot m\cdot b\cdot\langle s, c, t\rangle = \langle\, \mathit{mxinsdel}\cdot c\cdot b\cdot s \,,\, \mathit{c1} \,,\, \mathit{t1} \,\rangle$ is a proper choice for the case $b<c$, provided that $\mathit{c1}$ and $\mathit{t1}$ are defined by $\langle \mathit{t1}, \mathit{c1}\rangle = \mathit{insdelmn}\cdot m\cdot t$.

* * *

For the case $b=c$ we derive:

$$
\begin{array}{ll}
 & \llbracket\, \mathit{mxinsdel}\cdot m\cdot b\cdot\langle s, c, t\rangle \,\rrbracket \\
= & \quad \{ \text{ specification of } \mathit{mxinsdel} \ \} \\
 & \{m\} \cup (\llbracket\, \langle s, c, t\rangle \,\rrbracket \setminus \{b\}) \\
= & \quad \{ \text{ definition of } \llbracket\cdot\rrbracket \ \} \\
 & \{m\} \cup ((\llbracket s \rrbracket \cup \{c\} \cup \llbracket t \rrbracket) \setminus \{b\}) \\
= & \quad \{\ b=c \text{, so } \neg(b\in s) \text{ and } \neg(b\in t) : \text{ set calculus } \} \\
 & \{m\} \cup \llbracket s \rrbracket \cup \llbracket t \rrbracket \\
= & \quad \{ \text{ introduce } \mathit{c1} \text{ with } \mathit{c1} = \mathsf{min}\cdot(\{m\} \cup \llbracket t \rrbracket) \ \} \\
 & \llbracket s \rrbracket \cup \{\mathit{c1}\} \cup (\{m\} \cup \llbracket t \rrbracket) \setminus \{\mathit{c1}\} \\
= & \quad \{ \text{ introduce } \mathit{t1} \text{ with } \llbracket\, \mathit{t1} \,\rrbracket = (\{m\} \cup \llbracket t \rrbracket) \setminus \{\mathit{c1}\} \ \}
\end{array}
$$

$$
\begin{array}{ll}
 & [\![\, s \,]\!] \cup \{\, c1 \,\} \cup [\![\, t1 \,]\!] \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \} \\
 & [\![\, \langle\, s \,,\, c1 \,,\, t1 \,\rangle \,]\!] \ ,
\end{array}
$$

from which we conclude that $mxinsdel \cdot m \cdot b \cdot \langle s, c, t \rangle = \langle\, s \,,\, c1 \,,\, t1 \,\rangle$ is correct for the case $b = c$, where $c1$ and $t1$ are defined by $\langle t1, c1 \rangle = insdelmn \cdot m \cdot t$.

* * *

Finally, for the case $c < b$ we derive:

$$
\begin{array}{ll}
 & [\![\, mxinsdel \cdot m \cdot b \cdot \langle s, c, t \rangle \,]\!] \\
= & \quad \{ \text{ specification of } mxinsdel \ \} \\
 & \{m\} \cup ([\![\, \langle s, c, t \rangle \,]\!] \setminus \{b\}) \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \ \} \\
 & \{m\} \cup ((\, [\![s]\!] \cup \{c\} \cup [\![t]\!] \,) \setminus \{b\}) \\
= & \quad \{\, c < b \text{, so } \neg(b \in s) \text{ and } b \in t \text{: set calculus } \} \\
 & \{m\} \cup [\![s]\!] \cup \{c\} \cup (\, [\![t]\!] \setminus \{b\} \,) \\
= & \quad \{\, c < m \text{: more set calculus } \} \\
 & [\![s]\!] \cup \{c\} \cup \{m\} \cup (\, [\![t]\!] \setminus \{b\} \,) \\
= & \quad \{\ (\forall d : d \in t : d < m\,) \wedge b \in t \text{: specification of } mxinsdel \ \} \\
 & [\![s]\!] \cup \{c\} \cup [\![\, mxinsdel \cdot m \cdot b \cdot t \,]\!] \\
= & \quad \{ \text{ definition of } [\![\cdot]\!] \ \} \\
 & [\![\, \langle\, s \,, c \,, mxinsdel \cdot m \cdot b \cdot t \,\rangle \,]\!] \ ,
\end{array}
$$

so, $mxinsdel \cdot m \cdot b \cdot \langle s, c, t \rangle = \langle\, s \,, c \,, mxinsdel \cdot m \cdot b \cdot t \,\rangle$ is appropriate if $c < b$.

* * *

Combining the above three results we obtain this definition for $mxinsdel$:

$$
\begin{array}{l}
mxinsdel \cdot m \cdot b \cdot \langle s, c, t \rangle \ = \\
\qquad \mathsf{if} \ \ b < c \ \rightarrow \ \langle\, mxinsdel \cdot c \cdot b \cdot s \,,\, c1 \,,\, t1 \,\rangle \\
\qquad\qquad\qquad\quad \mathsf{whr} \ \langle t1, c1 \rangle = insdelmn \cdot m \cdot t \ \mathsf{end} \\
\qquad [\!] \ \ b = c \ \rightarrow \ \langle\, s \,,\, c1 \,,\, t1 \,\rangle \\
\qquad\qquad\qquad\quad \mathsf{whr} \ \langle t1, c1 \rangle = insdelmn \cdot m \cdot t \ \mathsf{end} \\
\qquad [\!] \ \ c < b \ \rightarrow \ \langle\, s \,, c \,, mxinsdel \cdot m \cdot b \cdot t \,\rangle \\
\qquad \mathsf{fi}
\end{array}
$$

5.3 mninsdel

We recall the specification of function *mxinsdel*, for all m, b, t:

$$(\forall c : c \in t : m < c) \ \wedge \ b \in t \ \Rightarrow$$
$$[\![\ mninsdel \cdot m \cdot b \cdot t \ ]\!] \ = \{ m \} \cup ([\![t]\!] \setminus \{ b \})$$

Function *mninsdel* is the symmetric counterpart of *mxinsdel*. By exploiting this symmetry we obtain the following definition for *mninsdel*:

$$\begin{array}{l} mninsdel \cdot m \cdot b \cdot \langle s, c, t \rangle \ = \\ \quad \mathsf{if} \ \ c < b \ \rightarrow \ \langle \, s1 \, , c1 \, , mninsdel \cdot c \cdot b \cdot t \, \rangle \\ \qquad\qquad\quad \mathsf{whr} \ \langle s1, c1 \rangle \ = \ insdelmx \cdot m \cdot s \ \mathsf{end} \\ \quad [] \ \ b = c \ \rightarrow \ \langle \, s1 \, , c1 \, , t \, \rangle \\ \qquad\qquad\quad \mathsf{whr} \ \langle s1, c1 \rangle \ = \ insdelmx \cdot m \cdot s \ \mathsf{end} \\ \quad [] \ \ b < c \ \rightarrow \ \langle \, mninsdel \cdot m \cdot b \cdot s \, , c \, , t \, \rangle \\ \quad \mathsf{fi} \end{array}$$

5.4 delmn

We recall the specification of function *delmn*, for all s, r, m:

$$s \neq \langle \, \rangle \ \wedge \ delmn \cdot s \ = \ \langle r, m \rangle \ \Rightarrow$$
$$[\![\, r \,]\!] \ = \ [\![\, s \,]\!] \setminus \{ m \} \ \wedge \ m = \mathsf{min} \cdot [\![\, s \,]\!]$$

In the very much the same way as for the previous functions, the following definition for *delmn* can be derived:

$$\begin{array}{lll} delmn \cdot \langle s, c, t \rangle \ = & \mathsf{if} \ \ s = \langle \, \rangle & \rightarrow \ \langle \langle \, \rangle \, , c \rangle \\ & [] \ \ s \neq \langle \, \rangle \wedge \# s = 1 + \# t & \rightarrow \ \langle \langle s1, c, t \rangle \, , m \, \rangle \\ & \quad \mathsf{whr} \ \langle s1, m \rangle \ = \ delmn \cdot s \ \mathsf{end} & \\ & [] \ \ s \neq \langle \, \rangle \wedge \# s = \# t & \rightarrow \ \langle \langle s1, m1, t1 \rangle \, , m \, \rangle \\ & \quad \mathsf{whr} \ \ \langle s1, m \rangle \ = \ insdelmn \cdot c \cdot s & \\ & \quad\ \ \& \ \langle t1, m1 \rangle \ = \ delmn \cdot t & \\ & \quad \mathsf{end} & \\ & \mathsf{fi} & \end{array}$$

5.5 delmx

We recall the specification of function *delmx* , for all s, r, m :

$$s \neq \langle\,\rangle \;\wedge\; delmx{\cdot}s = \langle r, m\rangle \;\Rightarrow$$
$$[\![\, r \,]\!] = [\![\, s \,]\!] \setminus \{\, m \,\} \;\wedge\; m = \mathsf{max}{\cdot}[\![\, s \,]\!]$$

In the very much the same way as for the previous functions, the following definition for *delmx* can be derived:

$$\begin{array}{lll}
delmx{\cdot}\langle s, c, t\rangle & = & \mathsf{if}\;\; t = \langle\,\rangle \qquad\qquad\qquad\quad \rightarrow\;\; \langle s, c\rangle \\
& & [\!]\;\; t \neq \langle\,\rangle \wedge \#s = \#t \quad \rightarrow\;\; \langle\, \langle s, c, t1\,\rangle, m\,\rangle \\
& & \quad \mathsf{whr}\; \langle t1, m\rangle \;=\; delmx{\cdot}t\; \mathsf{end} \\
& & [\!]\;\; t \neq \langle\,\rangle \wedge \#s = 1{+}\#t \;\rightarrow\;\; \langle\, \langle s1, m1, t1\,\rangle, m\,\rangle \\
& & \quad \mathsf{whr}\;\; \langle t1, m\rangle \;=\; insdelmx{\cdot}c{\cdot}t \\
& & \qquad \&\; \langle s1, m1\rangle \;=\; delmx{\cdot}s \\
& & \quad \mathsf{end} \\
& & \mathsf{fi}
\end{array}$$

6 Summary

Here we collect the specifications of all functions, and their definitions as derived in the previous sections. All type information is left implicit in the specifications: all tree parameters are *assumed to be*, and all tree function values are *required to be* ordered and balanced.

In the following we recapitulate the definition of the datatype, then we give the main functions – *isel*, *ins*, *del* –, and, finally, we list all auxiliary functions used, in alphabetical order.

6.0 the datatype

The basis is the datatype T of labelled binary trees, defined recursively by:

$$\langle\,\rangle \in T \;\text{ , and:}$$
$$\langle s, c, t\rangle \in T \;\text{ , for all } c \in A \text{ and } s, t \in T \;\text{ .}$$

Unary operator $\#$ maps trees to their sizes, in the obvious way:

$$\begin{array}{lcl}
\#\langle\,\rangle & = & 0 \\
\#\langle s, c, t\rangle & = & \#s + 1 + \#t
\end{array}$$

Now, datatype OB is the set defined by, for all s:

$$s \in OB \;\Leftrightarrow\; s \in T \;\wedge\; O{\cdot}s \;\wedge\; B{\cdot}s \quad ,$$

where predicates O and B are defined (recursively) by:

$$\begin{array}{lcl} O{\cdot}\langle\,\rangle & \Leftrightarrow & \mathsf{true} \\ O{\cdot}\langle s,c,t\rangle & \Leftrightarrow & (\forall b : b \in s : b < c) \;\wedge\; (\forall b : b \in t : c < b) \;\wedge\; O{\cdot}s \;\wedge\; O{\cdot}t \\ B{\cdot}\langle\,\rangle & \Leftrightarrow & \mathsf{true} \\ B{\cdot}\langle s,c,t\rangle & \Leftrightarrow & \#t \le \#s \;\wedge\; \#s \le 1 + \#t \;\wedge\; B{\cdot}s \;\wedge\; B{\cdot}t \end{array}$$

6.1 set representation

Every tree $s \in T$ represents a set $[\![\, s \,]\!]$, where the abstraction function $[\![\cdot]\!]$ is defined by:

$$\begin{array}{lcl} [\![\langle\,\rangle]\!] & = & \phi \\ [\![\langle s,c,t\rangle]\!] & = & [\![\, s \,]\!] \cup \{\, c \,\} \cup [\![\, t \,]\!] \end{array}$$

6.2 is-element-of

specification: for all b, s: $\mathit{isel}{\cdot}b{\cdot}s \;\Leftrightarrow\; b \in [\![\, s \,]\!]$
□
definition:

$$\begin{array}{lcl} \mathit{isel}{\cdot}b{\cdot}\langle\,\rangle & = & \mathsf{false} \\ \mathit{isel}{\cdot}b{\cdot}\langle s,c,t\rangle & = & \mathsf{if}\;\; b<c \;\rightarrow\; \mathit{isel}{\cdot}b{\cdot}s \\ & & \;[]\;\; b=c \;\rightarrow\; \mathsf{true} \\ & & \;[]\;\; c<b \;\rightarrow\; \mathit{isel}{\cdot}b{\cdot}t \\ & & \mathsf{fi} \end{array}$$

□

6.3 insert

specification: for all b, s: $[\![\, \mathit{ins}{\cdot}b{\cdot}s \,]\!] \;=\; \{\, b \,\} \cup [\![\, s \,]\!]$
□
definition:

$$\begin{array}{lcl} \mathit{ins}{\cdot}b{\cdot}s & = & \mathsf{if}\;\; \mathit{isel}{\cdot}b{\cdot}s \;\rightarrow\; s \\ & & \;[]\;\; \neg\, \mathit{isel}{\cdot}b{\cdot}s \;\rightarrow\; \mathit{inz}{\cdot}b{\cdot}s \\ & & \mathsf{fi} \end{array}$$

□

6.4 delete

specification: for all b, s: $[\![\, del \cdot b \cdot s \,]\!] \;=\; [\![\, s \,]\!] \setminus \{\, b \,\}$

□

definition:

$$
\begin{array}{lcl}
del \cdot b \cdot s & = & \mathsf{if}\ \neg\, isel \cdot b \cdot s \;\rightarrow\; s \\
& & [\!]\quad isel \cdot b \cdot s \;\rightarrow\; dlz \cdot b \cdot s \\
& & \mathsf{fi}
\end{array}
$$

□

6.5 delmn

specification: for all s, r, m:

$$s \neq \langle\,\rangle \;\wedge\; delmn \cdot s = \langle r, m \rangle \;\Rightarrow$$
$$[\![\, r \,]\!] = [\![\, s \,]\!] \setminus \{\, m \,\} \;\wedge\; m = \mathsf{min} \cdot [\![\, s \,]\!]$$

□

definition:

$$
\begin{array}{lcl}
delmn \cdot \langle s, c, t \rangle & = & \mathsf{if}\ \ s = \langle\,\rangle \qquad\qquad\qquad\quad \rightarrow\ \langle \langle\,\rangle, c \rangle \\
& & [\!]\ \ s \neq \langle\,\rangle \wedge \#s = 1 + \#t \ \rightarrow\ \langle \langle s1, c, t \rangle, m \rangle \\
& & \quad \mathsf{whr}\ \langle s1, m \rangle = delmn \cdot s\ \mathsf{end} \\
& & [\!]\ \ s \neq \langle\,\rangle \wedge \#s = \#t \qquad \rightarrow\ \langle \langle s1, m1, t1 \rangle, m \rangle \\
& & \quad \mathsf{whr}\ \ \langle s1, m \rangle = insdelmn \cdot c \cdot s \\
& & \qquad \&\ \langle t1, m1 \rangle = delmn \cdot t \\
& & \quad \mathsf{end} \\
& & \mathsf{fi}
\end{array}
$$

□

6.6 delmx

specification: for all s, r, m:

$$s \neq \langle\,\rangle \;\wedge\; delmx \cdot s = \langle r, m \rangle \;\Rightarrow$$
$$[\![\, r \,]\!] = [\![\, s \,]\!] \setminus \{\, m \,\} \;\wedge\; m = \mathsf{max} \cdot [\![\, s \,]\!]$$

□

definition:

$$
\begin{array}{lll}
delmx\cdot\langle s,c,t\rangle & = & \mathsf{if}\ \ t=\langle\,\rangle \qquad\qquad\qquad\quad \rightarrow\ \langle s,c\rangle \\
 & & [\!]\ \ t\neq\langle\,\rangle \wedge \#s=\#t \quad \rightarrow\ \langle\langle s,c,t1\rangle,m\rangle \\
 & & \quad \mathsf{whr}\ \langle t1,m\rangle = delmx\cdot t\ \mathsf{end} \\
 & & [\!]\ \ t\neq\langle\,\rangle \wedge \#s=1+\#t\ \rightarrow\ \langle\langle s1,m1,t1\rangle,m\rangle \\
 & & \quad \mathsf{whr}\ \ \langle t1,m\rangle = insdelmx\cdot c\cdot t \\
 & & \quad\ \ \&\ \langle s1,m1\rangle = delmx\cdot s \\
 & & \quad \mathsf{end} \\
 & & \mathsf{fi}
\end{array}
$$

□

6.7 dlz

specification: for all $b,s:\ b\in s\ \Rightarrow\ [\![\, dlz\cdot b\cdot s\,]\!] = [\![\, s\,]\!]\setminus\{b\}$

□

definition:

$$
\begin{array}{l}
dlz\cdot b\cdot\langle s,c,t\rangle\ = \\
\mathsf{if}\ \ b<c\wedge\#s=1+\#t\ \rightarrow\ \langle\, dlz\cdot b\cdot s\,,c\,,t\,\rangle \\
[\!]\ \ b<c\wedge\#s=\#t\quad\ \rightarrow\ \langle\, mxinsdel\cdot c\cdot b\cdot s\,,m\,,t1\,\rangle \\
\qquad\qquad\qquad\qquad\ \ \mathsf{whr}\ \langle t1,m\rangle = delmn\cdot t\ \mathsf{end} \\
[\!]\ \ b=c\wedge s=\langle\,\rangle\quad\ \rightarrow\ \langle\,\rangle \\
[\!]\ \ b=c\wedge\#s=\#t\wedge t\neq\langle\,\rangle\ \rightarrow\ \langle\, s\,,m\,,t1\,\rangle \\
\qquad\qquad\qquad\qquad\qquad\ \ \mathsf{whr}\ \langle t1,m\rangle = delmn\cdot t\ \mathsf{end} \\
[\!]\ \ b=c\wedge\#s=1+\#t\ \rightarrow\ \langle\, s1\,,m\,,t\,\rangle \\
\qquad\qquad\qquad\qquad\ \ \mathsf{whr}\ \langle s1,m\rangle = delmx\cdot s\ \mathsf{end} \\
[\!]\ \ c<b\wedge\#s=\#t\quad\ \rightarrow\ \langle\, s\,,c\,,dlz\cdot b\cdot t\,\rangle \\
[\!]\ \ c<b\wedge\#s=1+\#t\ \rightarrow\ \langle\, s1\,,m\,,mninsdel\cdot c\cdot b\cdot t\,\rangle \\
\qquad\qquad\qquad\qquad\ \ \mathsf{whr}\ \langle s1,m\rangle = delmx\cdot s\ \mathsf{end} \\
\mathsf{fi}
\end{array}
$$

□

6.8 insdelmn

specification: for all $b,s,r,m:$

$$
\begin{array}{l}
\neg(b\in s)\ \wedge\ insdelmn\cdot b\cdot s\ =\ \langle r,m\rangle\ \Rightarrow \\
[\![\, r\,]\!] = (\{b\}\cup[\![\, s\,]\!])\setminus\{m\}\ \ \wedge\ \ m=\mathsf{min}\cdot(\{b\}\cup[\![\, s\,]\!])
\end{array}
$$

□

definition:

$$
\begin{array}{lcl}
\mathit{insdelmn}\cdot b\cdot\langle\,\rangle & = & \langle\langle\,\rangle, b\rangle \\
\mathit{insdelmn}\cdot b\cdot\langle s,c,t\rangle & = & \langle\langle \mathit{s1}, \mathit{c1}, \mathit{t1}\rangle, m\,\rangle \\
 & & \quad \mathsf{whr}\ \langle \mathit{t1}, \mathit{c1}\rangle = \mathit{insdelmn}\cdot(b{\uparrow}c)\cdot t \\
 & & \quad\ \ \&\ \ \langle \mathit{s1}, m\rangle = \mathit{insdelmn}\cdot(b{\downarrow}c)\cdot s \\
 & & \quad \mathsf{end}
\end{array}
$$

□

6.9 insdelmx

specification: for all b, s, r, m:

$$\neg(b{\in}s)\ \wedge\ \mathit{insdelmx}\cdot b\cdot s = \langle r, m\rangle\ \Rightarrow$$

$$[\![\, r\,]\!] = (\{b\}\cup[\![\, s\,]\!])\setminus\{m\}\ \wedge\ m = \mathsf{max}\cdot(\{b\}\cup[\![\, s\,]\!])$$

□

definition:

$$
\begin{array}{lcl}
\mathit{insdelmx}\cdot b\cdot\langle\,\rangle & = & \langle\langle\,\rangle, b\rangle \\
\mathit{insdelmx}\cdot b\cdot\langle s,c,t\rangle & = & \langle\langle \mathit{s1}, \mathit{c1}, \mathit{t1}\rangle, m\,\rangle \\
 & & \quad \mathsf{whr}\ \langle \mathit{s1}, \mathit{c1}\rangle = \mathit{insdelmx}\cdot(b{\downarrow}c)\cdot s \\
 & & \quad\ \ \&\ \ \langle \mathit{t1}, m\rangle = \mathit{insdelmx}\cdot(b{\uparrow}c)\cdot t \\
 & & \quad \mathsf{end}
\end{array}
$$

□

6.10 inz

specification: for all b, s: $\neg(b\in s)\ \Rightarrow\ [\![\, \mathit{inz}\cdot b\cdot s\,]\!] = \{b\}\cup[\![\, s\,]\!]$

□

definition:

$$
\begin{array}{lcl}
\mathit{inz}\cdot b\cdot\langle\,\rangle & = & \langle\langle\,\rangle, b, \langle\,\rangle\rangle \\
\mathit{inz}\cdot b\cdot\langle s,c,t\rangle & = & \mathsf{if}\ \ b<c\wedge\#s=\#t\ \ \rightarrow\ \ \langle\, \mathit{inz}\cdot b\cdot s\,, c\,, t\,\rangle \\
 & & \ [\!]\ \ b<c\wedge\#s=1{+}\#t\ \rightarrow \\
 & & \quad \langle r\,, m\,, \mathit{inz}\cdot c\cdot t\,\rangle\ \ \mathsf{whr}\ \langle r, m\rangle = \mathit{insdelmx}\cdot b\cdot s\ \ \mathsf{end} \\
 & & \ [\!]\ \ c<b\wedge\#s=\#t\ \ \rightarrow \\
 & & \quad \langle \mathit{inz}\cdot c\cdot s\,, m\,, r\,\rangle\ \ \mathsf{whr}\ \langle r, m\rangle = \mathit{insdelmn}\cdot b\cdot t\ \ \mathsf{end} \\
 & & \ [\!]\ \ c<b\wedge\#s=1{+}\#t\ \rightarrow\ \ \langle\, s\,, c\,, \mathit{inz}\cdot b\cdot t\,\rangle \\
 & & \ \mathsf{fi}
\end{array}
$$

□

6.11 mninsdel

specification: for all m, b, s:

$$(\forall c : c \in s : m < c) \ \wedge \ b \in s \ \Rightarrow$$
$$[\![\, mninsdel \cdot m \cdot b \cdot s \,]\!] \ = \{ m \} \cup ([\![\, s \,]\!] \setminus \{ b \})$$

□

definition:

$$mninsdel \cdot m \cdot b \cdot \langle s, c, t \rangle \ =$$
$$\begin{array}{lll} \textsf{if} & c < b \ \rightarrow & \langle \, s1 \, , c1 \, , mninsdel \cdot c \cdot b \cdot t \, \rangle \\ & & \textsf{whr} \ \langle s1, c1 \rangle \ = \ insdelmx \cdot m \cdot s \ \textsf{end} \\ [\!] & b = c \ \rightarrow & \langle \, s1 \, , c1 \, , t \, \rangle \\ & & \textsf{whr} \ \langle s1, c1 \rangle \ = \ insdelmx \cdot m \cdot s \ \textsf{end} \\ [\!] & b < c \ \rightarrow & \langle \, mninsdel \cdot m \cdot b \cdot s \, , c \, , t \, \rangle \\ \textsf{fi} & & \end{array}$$

□

6.12 mxinsdel

specification: for all m, b, s:

$$(\forall c : c \in s : c < m) \ \wedge \ b \in s \ \Rightarrow$$
$$[\![\, mxinsdel \cdot m \cdot b \cdot s \,]\!] \ = \{ m \} \cup ([\![\, s \,]\!] \setminus \{ b \})$$

□

definition:

$$mxinsdel \cdot m \cdot b \cdot \langle s, c, t \rangle \ =$$
$$\begin{array}{lll} \textsf{if} & b < c \ \rightarrow & \langle \, mxinsdel \cdot c \cdot b \cdot s \, , c1 \, , t1 \, \rangle \\ & & \textsf{whr} \ \langle t1, c1 \rangle \ = \ insdelmn \cdot m \cdot t \ \textsf{end} \\ [\!] & b = c \ \rightarrow & \langle \, s \, , c1 \, , t1 \, \rangle \\ & & \textsf{whr} \ \langle t1, c1 \rangle \ = \ insdelmn \cdot m \cdot t \ \textsf{end} \\ [\!] & c < b \ \rightarrow & \langle \, s \, , c \, , mxinsdel \cdot m \cdot b \cdot t \, \rangle \\ \textsf{fi} & & \end{array}$$

□

Eindhoven, 27 january 2008

Rob R. Hoogerwoord
department of mathematics and computing science
Eindhoven University of Technology
postbus 513
5600 MB Eindhoven