Funcons-beta: Multisets *

The PLanCompS Project

Multisets.cbs | PLAIN | PRETTY

Multisets (bags)

> [*Type* multisets
> *Funcon* multiset
> *Funcon* multiset-elements
> *Funcon* multiset-occurrences
> *Funcon* multiset-insert
> *Funcon* multiset-delete
> *Funcon* is-submultiset]

> *Meta-variables* GT <: ground-values

> *Built-in Type* multisets(GT)

multisets(GT) is the type of possibly-empty finite multisets of elements of GT.

> *Built-in Funcon* multiset(_ : (GT)*) : $\Rightarrow$ multisets(GT)

Note that multiset($\cdots$) is not a constructor operation. The order of argument values is ignored, but duplicates are significant, e.g., multiset($1, 2, 2$) is equivalent to multiset($2, 1, 2$), but not to multiset($1, 2$) or multiset($2, 1$).

> *Built-in Funcon* multiset-elements(_ : multisets(GT)) : $\Rightarrow (GT)^*$

For each multiset MS, the sequence of values V^* returned by multiset-elements(MS) contains each element of MS the same number of times as MS does. The order of the values in V^* is unspecified, and may vary between multisets.

> *Assert* multiset(multiset-elements(S)) == S

> *Built-in Funcon* multiset-occurrences(_ : GT, _ : multisets(GT)) : $\Rightarrow$ natural-numbers

multiset-occurrences(GV, MS) returns the number of occurrences of GV in MS.

> *Built-in Funcon* multiset-insert(_ : GT, _ : natural-numbers, _ : multisets(GT)) : $\Rightarrow$ multisets(GT)

---

multiset-insert(GV, N, MS) returns the multiset that differs from MS by containing N more copies of GV.

multiset-delete($\_$: multisets(GT), $\_$: GT, $\_$: natural-numbers) : $\Rightarrow$ multisets(GT)

multiset-delete(MS, GV, N) removes N copies of V from the multiset MS, or all copies of GV if there are fewer than N in MS.

is-submultiset($\_$: multisets(GT), $\_$: multisets(GT)) : $\Rightarrow$ booleans

is-submultiset(MS_1, MS_2) tests whether every element of MS_1 has equal or fewer occurrences in MS_1 than in MS_2.