section number_toolkit

/Reference manual/Standard toolkit

1. Numbers

section *number_toolkit*

1.1. Successor

function($succ$ _)

$$\begin{array}{|l} succ\ \_ : \mathbb{P}(\mathbb{N} \times \mathbb{N}) \\ \hline (succ\ \_) = \lambda\, n : \mathbb{N} \bullet n + 1 \end{array}$$

The successor of a natural number n is equal to $n + 1$.

1.2. Integers

$$\mathbb{Z} : \mathbb{P}\,\mathbb{A}$$

$\mathbb{Z}$ is the set of integers, that is, positive and negative whole numbers and zero. The set $\mathbb{Z}$ is characterised by axioms for its additive structure given in the prelude together with the next formal paragraph below.

Number systems that extend the integers may be specified as supersets of $\mathbb{Z}$.

1.3. Addition of integers, arithmetic negation

function(-_)

$$
\begin{array}{|l}
\text{-}\_ : \mathbb{P}(\mathbb{A} \times \mathbb{A}) \\
\hline
\forall\, x, y : \mathbb{Z} \bullet \exists_1\, z : \mathbb{Z} \bullet ((x, y), z) \in (\_ + \_) \\
\forall\, x : \mathbb{Z} \bullet \exists_1\, y : \mathbb{Z} \bullet (x, y) \in (\text{-}\_) \\
\forall\, i, j, k : \mathbb{Z} \bullet \\
\qquad (i + j) + k = i + (j + k) \\
\qquad \wedge\ i + j = j + i \\
\qquad \wedge\ i + \text{-}i = 0 \\
\qquad \wedge\ i + 0 = i \\
\mathbb{Z} = \{ z : \mathbb{A} \mid \exists\, x : \mathbb{N} \bullet z = x \vee z = \text{-}x \}
\end{array}
$$

The binary addition operator $(\_ + \_)$ is defined in the prelude. The definition here introduces additional properties for integers. The addition and negation operations on integers are total functions that take integer values. The integers form a commutative group under $(\_ + \_)$ with - as the inverse operation and 0 as the identity element. NOTE If *function_toolkit* notation were exploited, the negation operator could be defined as follows.

$$
\begin{array}{|l}
\text{-}\_ : \mathbb{A} \leftrightarrow \mathbb{A} \\
\hline
(\mathbb{Z} \times \mathbb{Z}) \lhd (\_ + \_) \in \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z} \\
\mathbb{Z} \lhd \text{-} \in \mathbb{Z} \rightarrow \mathbb{Z} \\
\forall\, i, j, k : \mathbb{Z} \bullet \\
\qquad (i + j) + k = i + (j + k) \\
\qquad \land\ i + j = j + i \\
\qquad \land\ i + \text{-}i = 0 \\
\qquad \land\ i + 0 = i \\
\forall\, h : \mathbb{P}\,\mathbb{Z} \bullet \\
\qquad 1 \in h \land (\forall\, i, j : h \bullet i + j \in h \land \text{-}i \in h) \\
\qquad\qquad \Rightarrow h = \mathbb{Z}
\end{array}
$$

1.4. Subtraction

function 30 leftassoc$(_ - _)$

$$
\begin{array}{|l}
\_ - \_ : \mathbb{P}((\mathbb{A} \times \mathbb{A}) \times \mathbb{A}) \\
\hline
\forall\, x, y : \mathbb{Z} \bullet \exists_1\, z : \mathbb{Z} \bullet ((x, y), z) \in (\_ - \_) \\
\forall\, i, j : \mathbb{Z} \bullet i - j = i + \text{-}j
\end{array}
$$

Subtraction is a function whose domain includes all pairs of integers. For all integers i and j, $i - j$ is equal to $i + \text{-}j$. NOTE If *function_toolkit* notation were exploited, the subtraction operator could be defined as follows.

$$
\begin{array}{|l}
\_ - \_ : \mathbb{A} \times \mathbb{A} \leftrightarrow \mathbb{A} \\
\hline
(\mathbb{Z} \times \mathbb{Z}) \lhd (\_ - \_) \in \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z} \\
\forall i, j : \mathbb{Z} \bullet i - j = i + \text{-}j
\end{array}
$$

1.5. Less-than-or-equal

$$\text{relation}(\_ \leq \_)$$

$$
\begin{array}{|l}
\_ \leq \_ : \mathbb{P}(\mathbb{A} \times \mathbb{A}) \\
\hline
\forall i, j : \mathbb{Z} \bullet i \leq j \Leftrightarrow j - i \in \mathbb{N}
\end{array}
$$

For all integers i and j, $i \leq j$ if and only if their difference $j - i$ is a natural number.

1.6. Less-than

$$\text{relation}(\_ < \_)$$

$$
\begin{array}{|l}
\_ < \_ : \mathbb{P}(\mathbb{A} \times \mathbb{A}) \\
\hline
\forall\, i, j : \mathbb{Z} \bullet i < j \Leftrightarrow i + 1 \leq j
\end{array}
$$

For all integers i and j, $i < j$ if and only if $i + 1 \leq j$.

1.7. Greater-than-or-equal

relation$(_ \geq _)$

$$
\begin{array}{|l}
\_ \geq \_ : \mathbb{P}(\mathbb{A} \times \mathbb{A}) \\
\hline
\forall\, i, j : \mathbb{Z} \bullet i \geq j \Leftrightarrow j \leq i
\end{array}
$$

For all integers i and j, $i \geq j$ if and only if $j \leq i$.

1.8. Greater-than

relation$(_ > _)$

$$
\begin{array}{|l}
\_ > \_ : \mathbb{P}(\mathbb{A} \times \mathbb{A}) \\
\hline
\forall\, i, j : \mathbb{Z} \bullet i > j \Leftrightarrow j < i
\end{array}
$$

For all integers i and j, $i > j$ if and only if $j < i$.

Page 6 of 8

1.9. Strictly positive natural numbers

$$\mathbb{N}_1 == \{x : \mathbb{N} \mid \neg\, x = 0\}$$

The strictly positive natural numbers $\mathbb{N}_1$ are the natural numbers except zero.

1.10. Non-zero integers

$$\mathbb{Z}_1 == \{x : \mathbb{Z} \mid \neg\, x = 0\}$$

The non-zero integers $\mathbb{Z}_1$ are the integers except zero.

1.11. Multiplication of integers

$$\text{function } 40 \text{ leftassoc}(\_ * \_)$$

$$\begin{array}{|l}
_ * _ : \mathbb{P}((\mathbb{A} \times \mathbb{A}) \times \mathbb{A}) \\
\hline
\forall x, y : \mathbb{Z} \bullet \exists_1 z : \mathbb{Z} \bullet ((x, y), z) \in (_ * _) \\
\forall i, j, k : \mathbb{Z} \bullet \\
\qquad (i * j) * k = i * (j * k) \\
\qquad \land\ i * j = j * i \\
\qquad \land\ i * (j + k) = i * j + i * k \\
\qquad \land\ 0 * i = 0 \\
\qquad \land\ 1 * i = i
\end{array}$$

The binary multiplication operator $(\_ * \_)$ is defined for integers. The multiplication operation on integers is a total function and has integer values. Multiplication on integers is characterised by the unique operation under which the integers become a commutative ring with identity element 1. NOTE If *function_toolkit* notation were exploited, the multiplication operator could be defined as follows.

$$
\begin{array}{|l}
\_ * \_ : (\mathbb{A} \times \mathbb{A}) \nrightarrow \mathbb{A} \\
\hline
(\mathbb{Z} \times \mathbb{Z}) \lhd (\_ * \_) \in \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z} \\
\forall i, j, k : \mathbb{Z} \bullet \\
\quad\quad (i * j) * k = i * (j * k) \\
\quad\quad \land\ i * j = j * i \\
\quad\quad \land\ i * (j + k) = i * j + i * k \\
\quad\quad \land\ 0 * i = 0 \\
\quad\quad \land\ 1 * i = i
\end{array}
$$

1.12. Division, modulus

$$
\begin{array}{l}
\mathrm{function}\ 40\ \mathrm{leftassoc}(\_\ \mathit{div}\ \_) \\
\mathrm{function}\ 40\ \mathrm{leftassoc}(\_\ \mathit{mod}\ \_)
\end{array}
$$

$$
\begin{array}{|l}
\_\,div\,\_\,,\_\,mod\,\_ : \mathbb{P}((\mathbb{A} \times \mathbb{A}) \times \mathbb{A}) \\
\hline
\forall\, x : \mathbb{Z};\ y : \mathbb{Z}_1 \bullet \exists_1\, z : \mathbb{Z} \bullet ((x, y), z) \in (\_\,div\,\_) \\
\forall\, x : \mathbb{Z};\ y : \mathbb{Z}_1 \bullet \exists_1\, z : \mathbb{Z} \bullet ((x, y), z) \in (\_\,mod\,\_) \\
\forall\, i : \mathbb{Z};\ j : \mathbb{Z}_1 \bullet \\
\qquad i = (i\ div\ j) * j + i\ mod\ j \\
\qquad \land\ (0 \leq i\ mod\ j < j \lor j < i\ mod\ j \leq 0)
\end{array}
$$

For all integers i and non-zero integers j, the pair (i, j) is in the domain of $\_div\_$ and of $\_mod\_$, and $i\ div\ j$ and $i\ mod\ j$ have integer values.

When not zero, $i\ mod\ j$ has the same sign as j. This means that $i\ div\ j$ is the largest integer no greater than the rational number i/j. NOTE If *function_toolkit* notation were exploited, the division and modulus operators could be defined as follows.

$$
\begin{array}{|l}
\_\,div\,\_\,,\_\,mod\,\_ : \mathbb{A} \times \mathbb{A} \mathrel{+\!\!\!\!\rightarrow} \mathbb{A} \\
\hline
(\mathbb{Z} \times \mathbb{Z}_1) \lhd (\_\,div\,\_) \in \mathbb{Z} \times \mathbb{Z}_1 \rightarrow \mathbb{Z} \\
(\mathbb{Z} \times \mathbb{Z}_1) \lhd (\_\,mod\,\_) \in \mathbb{Z} \times \mathbb{Z}_1 \rightarrow \mathbb{Z} \\
\forall\, i : \mathbb{Z};\ j : \mathbb{Z}_1 \bullet \\
\qquad i = (i\ div\ j) * j + i\ mod\ j \\
\qquad \land\ (0 \leq i\ mod\ j < j \lor j < i\ mod\ j \leq 0)
\end{array}
$$

IT 22-Jan-2002