

# Symbols defined in the toolkit

[/Reference manual](#)

Here are the symbols defined in cadiz's extended toolkit, each with its  $\text{\LaTeX}$  and troff mark-up, the category of operator it denotes, and a longer name for the symbol. All symbols of the ISO Standard Z toolkit are present amongst these symbols, though the section structure differs.

## 1. section prelude

Symbol	$\text{\LaTeX}$	troff	category	name
$\mathbb{A}$	<code>\arithmos</code>	<code>arithmos</code>		type of all numbers
$\mathbb{N}$	<code>\nat</code>	<code>nn</code>		natural numbers
<i>succ</i>	<code>succ</code>	<code>succ</code>		successor function
$+$	<code>+</code>	<code>+</code>	function 30 leftassoc	addition
$*$	<code>*</code>	<code>*</code>	function 40 leftassoc	multiplication

## 2. section numdefs

Page 2 of 10



Go Back

Full Screen

Close

Quit

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
$\div$	<code>\divides</code>	<code>divides</code>	function 40 leftassoc	real division
$\mathbb{N}_1$	<code>\nat_1</code>	<code>nn^1</code>		positives
$\mathbb{Q}_+$	<code>\rat_+</code>	<code>fraction</code>		fractions
$\mathbb{Q}$	<code>\rat</code>	<code>rat</code>		rationals
$-$	<code>\negate</code>	<code>~</code>		integer negation
$-$	<code>-</code>	<code>-</code>	function 30 leftassoc	subtraction
$<$	<code>&lt;</code>	<code>&lt;</code>	relation (infix)	less than
$\mathbb{R}$	<code>\real</code>	<code>real</code>		reals
$\leq$	<code>\leq</code>	<code>&lt;=</code>	relation (infix)	less than or equal
$\geq$	<code>\geq</code>	<code>&gt;=</code>	relation (infix)	greater than or equal
$>$	<code>&gt;</code>	<code>&gt;</code>	relation (infix)	greater than
$\mathbb{Z}$	<code>\num</code>	<code>ints</code>		integers
$\operatorname{div}$	<code>\div</code>	<code>div</code>	function 40 leftassoc	division
$\operatorname{mod}$	<code>\mod</code>	<code>mod</code>	function 40 leftassoc	modulus

## 3. section fundefs

Page 3 of 10



Go Back

Full Screen

Close

Quit

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
$\neq$	<code>\neq</code>	<code>&lt;&gt;</code>	relation (infix)	inequality
$\notin$	<code>\notin</code>	<code>notmem</code>	relation (infix)	not member
$\emptyset$	<code>\emptyset</code>	<code>empty</code>		empty set
$\subseteq$	<code>\subseteq</code>	<code>sinc</code>	relation (infix)	subset
$\subset$	<code>\subset</code>	<code>ssinc</code>	relation (infix)	strict subset
$\mathbb{P}_1$	<code>\power_1</code>	<code>ps^1</code>	generic (prefix)	non-empty subsets
$\cup$	<code>\cup</code>	<code>sor</code>	function 30 leftassoc	set union
$\cap$	<code>\cap</code>	<code>sand</code>	function 40 leftassoc	set intersection
$\setminus$	<code>\setminus</code>	<code>sdiff</code>	function 30 leftassoc	set difference
$\ominus$	<code>\symdiff</code>	<code>symdiff</code>	function 25 leftassoc	symmetric set difference
$\bigcup$	<code>\bigcup</code>	<code>dor</code>		distributed set union
$\bigcap$	<code>\bigcap</code>	<code>dand</code>		distributed set intersection
$\mathbb{F}$	<code>\finset</code>	<code>fss</code>	generic (prefix)	finite subsets
$\mathbb{F}_1$	<code>\finset_1</code>	<code>fss^1</code>	generic (prefix)	non-empty finite subsets
$\leftrightarrow$	<code>\rel</code>	<code>rel</code>	generic 5 rightassoc	relation
$\mapsto$	<code>\mapsto</code>	<code>mlet</code>	function 10 leftassoc	maplet
<i>first</i>	<code>first</code>	<code>first</code>		project first from pair
<i>second</i>	<code>second</code>	<code>second</code>		project second from pair

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
$dom$	<code>\dom</code>	<code>dom</code>		domain
$ran$	<code>\ran</code>	<code>ran</code>		range
$id$	<code>\id</code>	<code>id</code>	generic (prefix)	identity
$\circ$	<code>\comp</code>	<code>frc</code>	function 40 leftassoc	forward relational composition
$\circ$	<code>\circ</code>	<code>rc</code>	function 40 leftassoc	relational composition
$\triangleleft$	<code>\dres</code>	<code>dres</code>	function 61 leftassoc	domain restriction
$\triangleright$	<code>\rres</code>	<code>rres</code>	function 60 leftassoc	range restriction
$\triangleleft$	<code>\ndres</code>	<code>dsub</code>	function 61 leftassoc	domain subtraction
$\triangleright$	<code>\nrres</code>	<code>rsub</code>	function 60 leftassoc	range subtraction
$\sim$	<code>\inv</code>	<code>inv</code>	function (postfix)	relational inverse
$( $	<code>\limg</code>	<code>opimg</code>	function (postfix)	open image bracket
$) $	<code>\rimg</code>	<code>climg</code>	function (postfix)	close image bracket
$upperBound$	<code>upperBound</code>	<code>upperBound</code>		upper bound
$disjoint$	<code>\disjoint</code>	<code>disjoint</code>	relation (prefix)	disjointness
$partition$	<code>\partition</code>	<code>partition</code>	relation (infix)	partitions
$\oplus$	<code>\oplus</code>	<code>fxov</code>	function 50 leftassoc	overriding



Page 8 of 10



Go Back

Full Screen

Close

Quit

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category
<i>transitive</i>	transitive	transitive	generic
<i>antisymmetric</i>	antisymmetric	antisymmetric	generic
<i>reflexive</i>	reflexive	reflexive	generic
<i>irreflexive</i>	irreflexive	irreflexive	generic
<i>preOrder</i>	preOrder	preOrder	generic
<i>order</i>	order	order	generic
<i>reflexiveOrder</i>	reflexiveOrder	reflexiveOrder	generic
<i>irreflexiveOrder</i>	irreflexiveOrder	irreflexiveOrder	generic
<i>totalOrder</i>	totalOrder	totalOrder	generic
<i>reflexiveTotalOrder</i>	reflexiveTotalOrder	reflexiveTotalOrder	generic
<i>irreflexiveTotalOrder</i>	irreflexiveTotalOrder	irreflexiveTotalOrder	generic
<i>reflexiveChain</i>	reflexiveChain	reflexiveChain	generic
<i>irreflexiveChain</i>	irreflexiveChain	irreflexiveChain	generic

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
+	<code>\plus</code>	<code>nrtc</code>	function (postfix)	transitive closure
*	<code>\star</code>	<code>rtc</code>	function (postfix)	reflexive transitive closure
<i>do</i>	<code>do</code>	<code>do</code>		repetition
$\rightarrow$	<code>\pfun</code>	<code>pfx</code>	generic 5 rightassoc	partial function
$\longrightarrow$	<code>\fun</code>	<code>fx</code>	generic 5 rightassoc	function
$\rightarrowtail$	<code>\pinj</code>	<code>pinj</code>	generic 5 rightassoc	partial injection
$\hookrightarrow$	<code>\inj</code>	<code>inj</code>	generic 5 rightassoc	injection
$\twoheadrightarrow$	<code>\psurj</code>	<code>psurj</code>	generic 5 rightassoc	partial surjection
$\twoheadrightarrow$	<code>\surj</code>	<code>surj</code>	generic 5 rightassoc	surjection
$\xrightarrow{\sim}$	<code>\bij</code>	<code>bij</code>	generic 5 rightassoc	bijection
$\rightrightarrows$	<code>\ffun</code>	<code>pffx</code>	generic 5 rightassoc	partial finite function
$\rightrightarrows$	<code>\finj</code>	<code>pfinj</code>	generic 5 rightassoc	partial finite injection

## 4. section seqdefs

Page 7 of 10



Go Back

Full Screen

Close

Quit

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
$\dots$	<code>\upto</code>	$\dots$	function 20 leftassoc	number interval
$\#$	<code>\#</code>	length	function (prefix)	cardinality
$items$	<code>items</code>	$items$		bag of elements of
$min$	<code>min</code>	$min$		minimum
$max$	<code>max</code>	$max$		maximum
$sequence$	<code>\sequence</code>	$sequence$	generic (prefix)	set of infinite sequen
$sequence_1$	<code>\sequence_1</code>	$sequence^1$	generic (prefix)	set of non-empty in
$isequence$	<code>\isequence</code>	$isequence$	generic (prefix)	set of injective infin
$isequence_1$	<code>\isequence_1</code>	$isequence^1$	generic (prefix)	set of non-empty in
$seq$	<code>\seq</code>	$seq$	generic (prefix)	set of finite sequen
$seq_1$	<code>\seq_1</code>	$seq^1$	generic (prefix)	set of non-empty fi
$iseq$	<code>\iseq</code>	$iseq$	generic (prefix)	set of injective finit
$iseq_1$	<code>\iseq_1</code>	$iseq^1$	generic (prefix)	set of injective non
$\langle$	<code>\langle</code>	$opseq$	function (nofix)	open sequence brack
$\rangle$	<code>\rangle</code>	$clseq$	function (nofix)	close sequence brack

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
$\sim$	<code>\cat</code>	<code>cat</code>	function 30 leftassoc	sequence concatenation
<i>rev</i>	<code>rev</code>	<code>rev</code>		reverse of sequence
<i>head</i>	<code>head</code>	<code>head</code>		project first from sequence
<i>last</i>	<code>last</code>	<code>last</code>		project last from sequence
<i>tail</i>	<code>tail</code>	<code>tail</code>		sequence without head
<i>front</i>	<code>front</code>	<code>front</code>		sequence without last
<i>squash</i>	<code>squash</code>	<code>squash</code>		compaction
$\uparrow$	<code>\filter</code>	<code>srres</code>	function 40 leftassoc	sequence filtering
$\downarrow$	<code>\extract</code>	<code>sres</code>	function 40 leftassoc	sequence extraction
<i>prefix</i>	<code>\prefix</code>	<code>prefix</code>	relation (infix)	sequence prefix relation
<i>suffix</i>	<code>\suffix</code>	<code>suffix</code>	relation (infix)	sequence suffix relation
<i>infix</i>	<code>\infix</code>	<code>in</code>	relation (infix)	sequence segment relation
$+/$	<code>\dsum</code>	<code>dsum</code>		distributed sum
$\sim/$	<code>\dcat</code>	<code>dcat</code>		distributed concatenation
<i>iter</i>	<code>iter</code>	<code>iter</code>		relational iteration
	<code>~{</code>	<code>iterup</code>	function (postfix)	begin superscripted relational
	<code>}</code>	<code>iterdown</code>	function (postfix)	end superscripted relational it



## 5. section bagkit

Page 9 of 10



Go Back

Full Screen

Close

Quit

Symbol	L <sup>A</sup> T <sub>E</sub> X	troff	category	name
bag	<code>\bag</code>	bag	generic (prefix)	set of bags
<i>count</i>	<code>count</code>	count		multiplicity
#	<code>\bcount</code>	bagoccurs	function 50 leftassoc	infix multiplicity
⊗	<code>\otimes</code>	bagscale	function 40 leftassoc	bag scaling
[	<code>\lbag</code>	opbag	function (nofix)	open bag bracket
]	<code>\rbag</code>	clbag	function (nofix)	close bag bracket
inbag	<code>\inbag</code>	bagmem	relation (infix)	bag membership
⊆	<code>\subbageq</code>	baginc	relation (infix)	sub-bag relation
⊕	<code>\uplus</code>	unionbag	function 30 leftassoc	bag union
⊖	<code>\uminus</code>	diffbag	function 30 leftassoc	bag difference

## 6. Toolkit operator precedences

Page 10 of 10

Go Back

Full Screen

Close

Quit

relation		$\neq \notin \subseteq \subset < \leq \geq >$ <i>prefix suffix</i> infix <i>partition</i> inbag $\sqsubseteq$ <i>disjoin</i>
generic	5	$\leftrightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow$
function	8	$\times$
function	10	$\mapsto$
function	20	$\dots$
function	25	$\ominus$
function	30	$+ - \cup \setminus \cap \uplus \sqcup$
function	40	$* \div \text{div mod } \bigcap \circ \circ \circ \circ \circ \circ \circ \circ$
function	50	$\oplus \#$
function	60	$\triangleright \triangleright$
function	61	$\triangleleft \triangleleft$
generic (prefix)		$\mathbb{P} \mathbb{P}_1 \text{id } \mathbb{F} \mathbb{F}_1$
generic (prefix)		<i>sequence sequence<sub>1</sub> issequence issequence<sub>1</sub> seq seq<sub>1</sub> iseq iseq<sub>1</sub> ba</i>
generic (prefix)		transitive antisymmetric reflexive irreflexive preOrder order
generic (prefix)		reflexiveOrder irreflexiveOrder totalOrder reflexiveTotalOrder
generic (prefix)		irreflexiveTotalOrder reflexiveChain irreflexiveChain
function (postfix)		$\sim \langle \rangle ^+ *$
function (nofix)		$\langle \rangle \llbracket \rrbracket$

IT 5-Jan-2002