

<i>function</i>	\Rightarrow	$\backslash\mathbf{begin} \left\{ \mathbf{eqcode} \right\} \left\{ \mathbf{id} \right\}$ $\left\{ [id [, id]^*] \right\}$ $\left\{ [ext_type [, ext_type]^*] \right\} \left\{ ext_type \right\}$ <i>instr_list</i> $\backslash\mathbf{end} \left\{ \mathbf{eqcode} \right\}$
<i>indexes</i>	\Rightarrow	$([upper] [lower] lower upper)$
<i>idx</i>	\Rightarrow	$\mathbf{id} \text{ indexes}$
<i>numx</i>	\Rightarrow	$(\mathbf{num} divide) \text{ indexes}$
<i>idx_numx</i>	\Rightarrow	$(idx numx)$
<i>upper</i>	\Rightarrow	$\wedge \left\{ ([(linear expr)] expr) \right\}$ $ $ $\wedge (\mathbf{id} \mathbf{num})$
<i>linear</i>	\Rightarrow	$(\backslash\mathbf{iter} [(+ -) \mathbf{num}] \mathbf{num})$
<i>lower</i>	\Rightarrow	$- \left\{ expr [, expr]^* \right\}$ $ $ $- (\mathbf{id} \mathbf{num})$
<i>type</i>	\Rightarrow	$\backslash\mathbf{type} \left\{ (\mathbf{Z} \mathbf{R} \mathbf{N} \mathbf{B}) \right\}$
<i>arraytype</i>	\Rightarrow	$\backslash\mathbf{arraytype} \left\{ (sexpr \backslash\mathbf{ldots}) [, (sexpr \backslash\mathbf{ldots})] \right\}$ $\left\{ (\mathbf{Z} \mathbf{R} \mathbf{N} \mathbf{B}) \right\}$
<i>ext_type</i>	\Rightarrow	$(type arraytype)$
<i>instr_list</i>	\Rightarrow	$[instr \backslash\mathbf{lend}]^*$
<i>instr</i>	\Rightarrow	<i>assign</i> $ $ <i>declare</i> $ $ <i>index_loop</i> $ $ <i>comment</i> $ $ <i>if_cond</i> $ $ <i>return</i>
<i>if_cond</i>	\Rightarrow	$\backslash\mathbf{qif} \left\{ cond_block \right\}$ <i>instr_list</i> $[\backslash\mathbf{qelseif} \left\{ cond_block \right\}$ <i>instr_list</i> $]^* [\backslash\mathbf{qelse}$ <i>instr_list</i> $]^* \backslash\mathbf{qendif}$
<i>cond_block</i>	\Rightarrow	$expr [comp \ expr]^+$ $[set_op \ expr [comp \ expr]^+]^*$
<i>assign</i>	\Rightarrow	$idx [, idx]^* \backslash\mathbf{gets} \ expr [, expr]^*$
<i>declare</i>	\Rightarrow	$idx [, idx]^* \backslash\mathbf{in} \ ext_type [, ext_type]^*$

<i>boolop</i>	\Rightarrow	$\backslash\text{land}$ $\backslash\text{lor}$ $\backslash\text{oplus}$
<i>binop</i>	\Rightarrow	$+$ $-$ $\backslash\text{cdot}$ $\backslash\text{ll}$ $\backslash\text{gg}$ $\backslash\text{mod}$
<i>divide</i>	\Rightarrow	$(\backslash\text{frac} \mid \backslash\text{dfrac}) \{ \text{expr} \} \{ \text{expr} \}$
<i>function_call</i>	\Rightarrow	$\backslash\text{call} \{ \text{id} \} \{ [\text{expr} [, \text{expr}]^*] \}$
<i>sexpr</i>	\Rightarrow	$(\backslash\text{not} \mid -) \text{sexpr_op} [(\text{binop} \mid \text{boolop}) \text{sexpr_op}]^*$ (expr) $\{ \text{expr} \}$
<i>sexpr_op</i>	\Rightarrow	$(\text{idx_numx} \mid \text{function_call} \mid \text{matrix})$
<i>filter</i>	\Rightarrow	$\backslash\text{filter} \{ \text{id} ^ \{ [\text{id}] \} $ $[, \text{id} ^ \{ [\text{id}] \}]^*$ $\mid \text{generator} \}$
<i>genarray</i>	\Rightarrow	$\backslash\text{genar} \backslash\text{limits} ^ \{ \text{expr} \} (\text{expr})$
<i>matrix</i>	\Rightarrow	$\backslash\text{begin} \{ \text{tmatrix} \}$ $[\text{expr} [\& \text{expr}]^* \backslash\text{lend}] +$ $\backslash\text{end} \{ \text{tmatrix} \}$
<i>expr</i>	\Rightarrow	$(\text{sexpr} \mid \text{filter} \mid \text{genarray}) \text{indexes}$
<i>index_loop</i>	\Rightarrow	$\text{idx} \mid \text{generator} \backslash\text{gets} (\text{expr} \mid \text{index_loop_cases})$
<i>index_loop_cases</i>	\Rightarrow	$\backslash\text{begin} \{ \text{cases} \}$ $[\text{expr} \& \text{generator}]^+$ $[\text{expr} \& \backslash\text{otherwise}] +$ $\backslash\text{end} \{ \text{cases} \}$

<i>print</i>	\Rightarrow	<code>\print { <i>expr</i> }</code>
<i>return</i>	\Rightarrow	<code>\return { <i>expr</i> }</code>
<i>generator</i>	\Rightarrow	<code>\forallall <i>id</i> [, <i>id</i>]*</code> <code><i>id</i> [, <i>id</i>]* : <i>cond_block</i></code>
<i>comp</i>	\Rightarrow	<code><</code> <code>></code> <code>\leq</code> <code>\geq</code> <code>[\textbf{not}] =</code>
<i>set_op</i>	\Rightarrow	<code>(\textbf{cup} \textbf{cap})</code>