function ⇒ **\begin** { **eqcode** } { ***id*** }
{ *[idx [* **,** *idx]*]* }
{ *[ext_type [* **,** *ext_type]*]* } { *ext_type* }
instr_list
\end { **eqcode** }

idx ⇒ ***id*** *[upper] [lower]*
| **num**

upper ⇒ ^ { *(* [*linear*] | *linear)* }

linear ⇒ ***id*** *[(* + | - *)* ***num*** *]*
| **num**

lower ⇒ _ { *sexpr [* **,** *sexpr]** }

type ⇒ **\type** { *(* **Z** | **R** | **N** | **B** *)* }

ext_type ⇒ *type [* ^ { ***num*** }
[_ { ***num*** *[* **,** ***num*** *]** } *]]*

instr_list ⇒ *[instr* **\endl** *]**

instr ⇒ *definition*
| *declaration*
| *with_loop*
| *return*

definition ⇒ *[idx]+ expr*

boolop ⇒ **\land**
| **\lor**
| **\oplus**

binop ⇒ +
| -
| **\cdot**
| *divide*
| **\ll**
| **\gg**
| **\mod**

divide ⇒ *(* **\frac** | **\dfrac** *)* **{** *expr* **}** **{** *expr* **}**

function_call ⇒ **\call** **{** ***id*** **}** **{** *[idx [* **,** *idx]*]* **}**

sexpr ⇒ *(* **\lnot** | **-** *) (* ***idx*** | *function_call) [(binop* | *boolop) (**idx*** | *function_call)]**

| **(** *sexpr* **)**

filter ⇒ **\filter** **{** ***id*** **^** **{** **[** ***id*** **]** **}** *[* **,** ***id*** **^** **{** **[** ***id*** **]** **}** *]** **|** *extended_condition* **}**

genarray ⇒ **\genar** **\limits** **^** **{** *expr* **}** **(** *sexpr* **)**

vector ⇒ **\begin** **{** **tvector** **}** *[sexpr* **\endl** *]+* **\end** **{** **tvector** **}**

matrix ⇒ **\begin** **{** **tmatrix** **}** **{** *[* ***id*** *]+* **}** *[sexpr [sexpr* **&** *]** **\endl** *]+* **\end** **{** **tmatrix** **}**

expr ⇒ *sexpr*

| *filter*

| *genarray*

| *vector*

| *matrix*

with_loop ⇒ *with_loop_wbr*

| *with_loop_wobr*

with_loop_wbr ⇒ *idx* **|** *extended_condition* **=** **\begin** **{** **cases** **}** *[expr* **&** *extended_condition]+* *[expr* **&** **\otherwise** *]+* **\end** **{** **cases** **}**

with_loop_wobr ⇒ *idx* **|** *extended_condition* **=** *expr*

return ⇒ **\return** **{** *expr* **}**