

# Grammar Of Starlet

$\langle \text{program} \rangle ::= \text{program id } \langle \text{block} \rangle \text{ endprogram}$

$\langle \text{block} \rangle ::= \langle \text{declarations} \rangle \langle \text{subprograms} \rangle \langle \text{statements} \rangle$

$\langle \text{declarations} \rangle ::= ( \text{declare } \langle \text{varlist} \rangle ; )^*$

$\langle \text{varlist} \rangle ::= \epsilon \mid \text{id } ( , \text{id } )^*$

$\langle \text{subprograms} \rangle ::= ( \langle \text{subprogram} \rangle )^*$

$\langle \text{subprogram} \rangle ::= \text{function id } \langle \text{funcbody} \rangle \text{ endfunction}$

$\langle \text{funcbody} \rangle ::= \langle \text{formalpars} \rangle \langle \text{block} \rangle$

$\langle \text{formalpars} \rangle ::= ( \langle \text{formalparlist} \rangle )$

$\langle \text{formalparlist} \rangle ::= \langle \text{formalparitem} \rangle ( , \langle \text{formalparitem} \rangle )^* \mid \epsilon$

$\langle \text{formalparitem} \rangle ::= \text{in id} \mid \text{inout id} \mid \text{inandout id}$

$\langle \text{statements} \rangle ::= \langle \text{statement} \rangle ( ; \langle \text{statement} \rangle )^*$

$\langle \text{statement} \rangle ::= \epsilon \mid$

$\langle \text{assignment-stat} \rangle \mid$

$\langle \text{if-stat} \rangle \mid$

$\langle \text{while-stat} \rangle \mid$

$\langle \text{do-while-stat} \rangle \mid$

$\langle \text{loop-stat} \rangle \mid$

$\langle \text{exit-stat} \rangle \mid$

$\langle \text{forcase-stat} \rangle \mid$

$\langle \text{incase-stat} \rangle \mid$

$\langle \text{return-stat} \rangle \mid$

$\langle \text{input-stat} \rangle \mid$

$\langle \text{print-stat} \rangle$

$\langle \text{assignment-stat} \rangle ::= \text{id} := \langle \text{expression} \rangle$

<if-stat> ::= if (<condition>) then <statements> <elsepart> endif

<elsepart> ::=  $\epsilon$  | else <statements>

<while-stat> ::= while ( <condition> ) <statements> endwhile

<do-while-stat> ::= dowhile <statements> enddowhile ( <condition> )

<loop-stat> ::= loop <statements> endloop

<exit-stat> ::= exit

<forcase-stat> ::= forcase  
  ( when ( <condition> ) : <statements> ) \*  
  default : <statements> enddefault  
endforcase

<incase-stat> ::= incase  
  ( when ( <condition> ) : <statements> ) \*  
endincase

<return-stat> ::= return <expression>

<print-stat> ::= print <expression>

<input-stat> ::= input id

<actualpars> ::= ( <actualparlist> )

<actualparlist> ::= <actualparitem> ( , <actualparitem> ) \* |  $\epsilon$

<actualparitem> ::= in <expression> | inout id | inandout id

<condition> ::= <boolterm> ( or <boolterm> ) \*

<boolterm> ::= <boolfactor> ( and <boolfactor> ) \*

<boolfactor> ::= not [ <condition> ] | [ <condition> ] |

<expression> <relational-oper> <expression>

$\langle \text{expression} \rangle ::= \langle \text{optional-sign} \rangle \langle \text{term} \rangle ( \langle \text{add-oper} \rangle \langle \text{term} \rangle )^*$

$\langle \text{term} \rangle ::= \langle \text{factor} \rangle ( \langle \text{mul-oper} \rangle \langle \text{factor} \rangle )^*$

$\langle \text{factor} \rangle ::= \text{constant} \mid ( \langle \text{expression} \rangle ) \mid \text{id} \langle \text{idtail} \rangle$

$\langle \text{idtail} \rangle ::= \epsilon \mid \langle \text{actualpars} \rangle$

$\langle \text{relational-oper} \rangle ::= = \mid <= \mid >= \mid > \mid < \mid <>$

$\langle \text{add-oper} \rangle ::= + \mid -$

$\langle \text{mul-oper} \rangle ::= * \mid /$

$\langle \text{optional-sign} \rangle ::= \epsilon \mid \langle \text{add-oper} \rangle$