|  |  |  |
| --- | --- | --- |
| *<program>* | ***->*** | **program** <identifier>  <declaration>\* <statement>\*  **stop**  **end** |
| *<declaration>* | ***->*** | <type> <identifier>  ((, <identifier>)\*  | ([(]<identifier> (, <identifier>)\*[)]))\* |
| *<type>* | ***->*** | **integer** | **real** | **logical** | **character** |
| *<line>* | ***->*** | <int-literal>? ([+] | [&])?  (<declaration> | <statement>)? |
| *<statement>* | ***->*** | <assignment>  | <parameter>  | <read>  | <write>  | <condition>  | <do-loop>  | <goto>  | <simple-expr> |
| *<assignment>* | ***->*** | <identifier> **=** <expression> |
| *<parameter>* | ***->*** | **parameter** [(]<assignment>[)] |
| *<read>* | ***->*** | **read** (\*, \*) <identifier>+ |
| *<write>* | ***->*** | **write** (\*, \*) <expression> |
| *<condition>* | ***->*** | **if** <expression> **then**? <statement>\*  (**elseif** <expression> **then**? <statement>\*)\*  (**else** <statement>\*)?  **endif** |
| *<do-loop>* | ***->*** | **do** <label> <assignment>, <expression>+  <statement>\*  <label> **continue** |
| *<goto>* | ***->*** | **goto** <label> |
| *<expression>* | ***->*** | <simple-expr> (<or-expr>)\* |
| *<or-expr>* | ***->*** | <and-expr> (**.or.** <and-expr>)? |
| *<and-expr>* | ***->*** | <equality-expr> (**.and.** <equality-expr>)? |
| *<equality-expr>* | ***->*** | <comp-expr> ((**.eq.** | **.ne.**) <comp-expr>)? |
| *<comp-expr>* | ***->*** | <add-expr>  ((**.ge.**|**.gt.**|**.le.**|**.lt.**) <add-expr>)? |
| *<add-expr>* | ***->*** | <mult-expr> ([**+ -**] <mult-expr>)\* |
| *<mult-expr>* | ***->*** | <pow-expr> ([**\* /**] <pow-expr>)\* |
| *<pow-expr>* | ***->*** | <neg-expr> (**\*\*** <pow-expr>)? |
| *<neg-expr>* | ***->*** | (**.not.** | **[-]**)? <simple-expr> |
| *<simple-expr>* | ***->*** | (**call**)? <identifier>  ([(](<expression>(, <expression>)\*)?[)])?  | <int-literal> | <logic-literal>  | <real-literal> | <string-literal>  | ( <expression> ) |
| *<function>* | ***->*** | <type> **function** <identifier>  ([(](<expression>(, <expression>)\*)?[)])  <declaration>\*  <statement>\*  **return**  **end** |
| *<subroutine>* | ***->*** | **subroutine** <identifier>  ([(](<expression>(, <expression>)\*)?[)])  <declaration>\*  <statement>\*  **return**  **end** |