|  |  |  |
| --- | --- | --- |
| **Pattern** | **Token** | **Purpose** |
| program | PROGRAM | Keyword |
| declare | DECLARE | Keyword |
| list | LIST | Keyword |
| of | OF | Keyword |
| variables | VARIABLES | Keyword |
| array | ARRAY | Simple array |
| size | SIZE | Keyword |
| values | VALUES | Keyword |
| jagged array | JAGARR | Jagged Array with different row sizes |
| integer | INTEGER | Keyword int |
| real | REAL | Keyword real |
| boolean | BOOLEAN | Keyword boolean |
| R1 | ROW1 | Keyword |
| ( | RBOP | Open Parenthesis |
| ) | RBCL | Closed Parenthesis |
| { | CBOP | Curly Open |
| } | CBCL | Curly Closed |
| [ | SQOP | Left Square Bracket |
| ] | SQCL | Right Square Bracket |
| : | COLON | Colon as Separator |
| ; | SEMICOLON | Semicolon as Separator |
| + | PLUS | Addition Operator |
| - | MINUS | Subtraction Operator |
| \* | MULT | Multiplication Operator |
| / | DIV | Division Operator |
|  | ASSIGN | Assignment Operator |
| &&& | AND | Logical and |
| ||| | OR | Logical or |
| .. | TWODOT | Range operator |

**Types:** integer, real, bool, array-> rectangular, jagged

**Variable Name:** alphabet, digits, underscore. Not start with digit

**Keywords:** program: PROGRAM, declare: DECLARE, list: LIST, of: OF , variables: VARIABLES, array: ARRAY, size: SIZE, values: VALUES, jagged array: JAGARR, of: OF, integer: INTEGER, real: REAL, boolean: BOOLEAN

**Symbols:** ( : RBOP , ) : RBCL , { : CBOP , }: CBCL , : : COLON , ; : SEMICOLON, [ : SQOP , ] : SQCL

**Arithmetic operators:** +: PLUS, -: MINUS, \*: MULT, /: DIV

**Boolean operators:** &&&: AND, |||: OR

No relational operators

//Start of program

<S> => PROGRAM RBOP RBCL CBOP <LIST\_OF\_STATEMENTS> CBCL

//Program is a list of declaration and assignment statements

<LIST\_OF\_STATEMENTS> => <DECLARATION\_STATEMENTS> <ASSIGNMENT\_STATEMENTS>

**DECLARATION STATEMENTS**

// Declaration statements is a list of declarations

<DECLARATION\_STATEMENTS> => <DECLARATION> <DECLARATION\_STATEMENTS> | <DECLARATION>

// Declarations can be of two type -> declaring single variable or declaring multiple variables

<DECLARATION> => <PRIMITIVE\_DECL> | <JAGGED\_ARRAY\_DECL>

// Declaration statement for primitive variables

<PRIMITIVE\_DECL> => DECLARE <DECLARE\_VARS> COLON <DECLARATION\_TYPE> SEMICOLON

// Declaration statement for jagged array variables

<JAGGED\_ARRAY\_DECL> => DECLARE <DECLARE\_VARS> COLON <DECLARE\_JAGGED>

// Declare single variable or a list of variables

<DECLARE\_VARS> => <ID1> | <LIST OF VARIABLES> <VAR\_NAME\_LIST>

// variable name list

<VAR\_NAME\_LIST> => <ID1> <VAR\_NAME\_LIST>| <ID1>

// Index of array

<IDX> => <ID1> | NUM | ARRAY\_ELE

<ARRAY\_ELE> => <ID1> SBOP <LIST> SBCL

<LIST> => <LIST> <IDX> | <IDX>

// Changes have to be made to account for variables with length at most 20.

// ID1 is a string which does not start with a digit

<ID1> => <LETTER> <ID2>

// ID2 is a string which can be a combination of letters and digits

<ID2> => <LETDIG> <ID2> | <LETDIG>

<LETDIG> => <DIGIT> | <LETTER> | EPSILON

<LETTER> => a|b|c ...............................Z | \_

<DIGIT> => 0|1|2...............|9

<DECLARATION\_TYPE> => <PRIMITIVE> | <DECLARE\_RECT\_ARRAY>

// Primitive are further categorized into 3 types

<PRIMITIVE> => INTEGER | REAL | BOOLEAN

// Rectangular array

<DECLARE\_RECT\_ARRAY> => ARRAY <ARRAY\_DIM> OF INTEGER

// Array dimensions are [num .. num][num .. num]....[num .. num]

<ARRAY\_DIM> => SBOP <IDX> TWODOT <IDX> SBCL <ARRAY\_DIM> | SBOP <IDX> TWODOT <IDX> SBCL

// Jagged array - 2D or 3D

<DECLARE\_JAGGED> => <DECLARE\_TWOD\_JAGGED> | <DECLARE\_THREED\_JAGGED>

// 2D [num .. num][ ]

<DECLARE\_TWOD\_JAGGED> => JAGARR SBOP <IDX> TWODOT <IDX> SBCL SBOP SBCL OF INTEGER SEMICOLON <TWOD\_JAGGED\_STATEMENTS>

<TWOD\_JAGGED\_STATEMENTS> => <TWOD\_JAGGED\_STATEMENT> <TWOD\_JAGGED\_STATEMENTS> | <TWOD\_JAGGED\_STATEMENT>

<TWOD\_JAGGED\_STATEMENT> => R SBOP INTEGER SBCL COLON SIZE INTEGER COLON VALUES CBOP <TWOD\_VALUES> CBCL

<TWOD\_VALUES> => <LIST> SEMICOLON <TWOD\_VALUES> | <LIST>

// 3D [num .. num][ ][ ]

<DECLARE\_THREED\_JAGGED> => JAGARR SBOP <IDX> TWODOT <IDX> SBCL SBOP SBCL SBOP SBCL OF INTEGER SEMICOLON <THREED\_JAGGED\_STATEMENTS>

<THREED\_JAGGED\_STATEMENTS>=><THREED\_JAGGED\_STATEMENT> <THREED\_JAGGED\_STATEMENTS> | <THREED\_JAGGED\_STATEMENT>

<THREED\_JAGGED\_STATEMENT>=>R SBOP INTEGER SBCL COLON SIZE INTEGER COLON VALUES CBOP

<THREED\_VALUES> CBCL

<THREED\_VALUES> => <LIST> SEMICOLON <THREED\_VALUES> | <LIST>

**ASSIGNMENT STATEMENTS**

// Assignment statements are list of assignments

<ASSIGNMENT\_STATEMENTS> => <ASSIGNMENT> <ASSIGNMENT\_STATEMENTS> | <ASSIGNMENT>

// An assignment -> lhs = rhs

<ASSIGNMENT> => <LHS> EQUALS <RHS> SEMICOLON

// LHS is basically a variable or an array element

<LHS> => <ID1> | <ARRAY\_ELE>

// Right hand expression can be an arithmetic expression or a boolean expression

<RHS> => <ARITHMETIC\_EXPR> | <BOOLEAN\_EXPR>

// grammar for arithmetic expressions

<ARITHMETIC\_EXPR> => <ARITHMETIC\_TERM> OP1 <ARITHMETIC\_EXPR> | <ARITHMETIC\_TERM>

<ARITHMETIC\_TERM> => IDX OP2 <ARITHMETIC\_TERM> | IDX

<OP1> => PLUS | MINUS

<OP2>=> MULT | DIV

// grammar for boolean expressions

<BOOLEAN\_EXPR> => <BOOLEAN\_TERM> OR <BOOLEAN\_EXPR> | <BOOLEAN\_TERM>

<BOOLEAN\_TERM> => <BOOLEAN\_FACTOR> AND <BOOLEAN\_TERM> | <BOOLEAN\_FACTOR>

<BOOLEAN\_FACTOR> => BOOL | <ID1>