Lexeme	Token	Attribute
program	PROGRAM_KW	-
globaltest	ID	Symbol Table Entry
int	INTEGER_KW	-
a	ID	Symbol Table Entry
;	SEMICOLON_KW	-
int	INTEGER_KW	-
b	ID	Symbol Table Entry
:=	ASSIGN_KW	-
#3	NUMCONST	-
;	SEMICOLON_KW	-
real	REAL_KW	-
f	ID	Symbol Table Entry
;	SEMICOLON_KW	-
real	REAL_KW	-
k	ID	Symbol Table Entry
:=	ASSIGN_KW	-
#0.4	REALCONST	-
;	SEMICOLON_KW	-
real	REAL_KW	-
l	ID	Symbol Table Entry
:=	ASSIGN_KW	-
#3.5	REALCONST	-
;	SEMICOLON_KW	-
char	CHARACTER_KW	-
c	ID	Symbol Table Entry
:=	ASSIGN_KW	-
'w'	CHARCONST	-
;	SEMICOLON_KW	-
char	CHARACTER_KW	-
h	ID	Symbol Table Entry
;	SEMICOLON_KW	-
bool	BOOLEAN_KW	-

s	ID	Symbol Table Entry
;	SEMICOLON_KW	-
bool	BOOLEAN_KW	-
g	ID	Symbol Table Entry
:=	ASSIGN_KW	-
TRUE	BOOLCONST	-
;	SEMICOLON_KW	-
int	INTEGER_KW	-
array	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
#2	NUMCONST	-
]	CLOSEBRACKET_KW	-
:=	ASSIGN_KW	-
{	OPENACCOLADE_KW	-
#1	NUMCONST	-
,	COMMA_KW	-
#7	NUMCONST	-
}	CLOSEACCOLADE_KW	-
;	SEMICOLON_KW	-
char	CHARACTER_KW	-
chars	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
.	DOT_KW	-
.	DOT_KW	-
]	CLOSEBRACKET_KW	-
:=	ASSIGN_KW	-
{	OPENACCOLADE_KW	-
'c'	CHARCONST	-
,	COMMA_KW	-
'd'	CHARCONST	-
,	COMMA_KW	-
'7'	CHARCONST	-
}	CLOSEACCOLADE_KW	-

;	SEMICOLON_KW	-
procedure	PROCEDURE_KW	-
p	ID	Symbol Table Entry
(	OPENPARENTHESIS_KW	-
int	INTEGER_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
char	CHARACTER_KW	-
t	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
{	OPENACCOLADE_KW	-
switch	SWITCH_KW	-
w	ID	Symbol Table Entry
case	CASE_KW	-
#10	NUMCONST	-
:	COLON_KW	-
{	OPENACCOLADE_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
+	ADD_KW	-
(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
w	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
-	SUB_KW	-
(	OPENPARENTHESIS_KW	-
+	ADD_KW	-
(	OPENPARENTHESIS_KW	-
*	MUL_KW	-

(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
w	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
w	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
w	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-
case	CASE_KW	-
#20	NUMCONST	-
:	COLON_KW	-
{	OPENACCOLADE_KW	-
t	ID	Symbol Table Entry
:=	ASSIGN_KW	-
and	AND_KW	-
(	OPENPARENTHESIS_KW	-
t	ID	Symbol Table Entry
,	COMMA_KW	-
t	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
or	OR_KW	-
(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
w	ID	Symbol Table Entry

)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
do	DO_KW	-
for	FOR_KW	-
i	ID	Symbol Table Entry
:=	ASSIGN_KW	-
#1	NUMCONST	-
upto	UPTO_KW	-
#10	NUMCONST	-
do	DO_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
%	MOD_KW	-
(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
#32	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
while	WHILE_KW	-
<>	NEQ_KW	-
(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
#1	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-
case	CASE_KW	-
#30	NUMCONST	-
:	COLON_KW	-
{	OPENACCOLADE_KW	-
if	IF_KW	-

not	NOT_KW	-
=	EQ_KW	-
(	OPENPARENTHESIS_KW	-
w	ID	Symbol Table Entry
,	COMMA_KW	-
#4	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
then	THEN_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
;	SEMICOLON_KW	-
else	ELSE_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
#9	NUMCONST	-
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-
default	DEFAULT_KW	-
:	COLON_KW	-
{	OPENACCOLADE_KW	-
if	IF_KW	-
and	AND_KW	-
(	OPENPARENTHESIS_KW	-
t	ID	Symbol Table Entry
,	COMMA_KW	-
t	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
and	AND_KW	-
then	THEN_KW	-
or	OR_KW	-
(	OPENPARENTHESIS_KW	-
t	ID	Symbol Table Entry
,	COMMA_KW	-

#0	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
then	THEN_KW	-
w	ID	Symbol Table Entry
:=	ASSIGN_KW	-
-	SUB_KW	-
w	ID	Symbol Table Entry
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-
end	END_KW	-
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-
;	SEMICOLON_KW	-
main	MAIN_KW	-
{	OPENACCOLADE_KW	-
array	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
#2	NUMCONST	-
]	CLOSEBRACKET_KW	-
:=	ASSIGN_KW	-
array	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
#2	NUMCONST	-
]	CLOSEBRACKET_KW	-
-	SUB_KW	-
(	OPENPARENTHESIS_KW	-
char	CHARACTER_KW	-
[	OPENBRACKET_KW	-
#2	NUMCONST	-
]	CLOSEBRACKET_KW	-
=	EQ_KW	-
'd'	CHARCONST	-
)	CLOSEPARENTHESIS_KW	-

;	SEMICOLON_KW	-
k	ID	Symbol Table Entry
:=	ASSIGN_KW	-
and	AND_KW	-
(	OPENPARENTHESIS_KW	-
or	OR_KW	-
else	ELSE_KW	-
(	OPENPARENTHESIS_KW	-
+	ADD_KW	-
(	OPENPARENTHESIS_KW	-
*	MUL_KW	-
(	OPENPARENTHESIS_KW	-
#2	NUMCONST	-
,	COMMA_KW	-
k	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
FALSE	BOOLCONST	-
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
TRUE	BOOLCONST	-
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
array	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
#1	NUMCONST	-
]	CLOSEBRACKET_KW	-
)	CLOSEPARENTHESIS_KW	-
,	COMMA_KW	-
#1	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
return	RETURN_KW	-

-	SUB_KW	-
(	OPENPARENTHESIS_KW	-
l	ID	Symbol Table Entry
,	COMMA_KW	-
k	ID	Symbol Table Entry
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
exit	EXIT_KW	-
when	WHEN_KW	-
not	NOT_KW	-
(	OPENPARENTHESIS_KW	-
>	GT_KW	-
(	OPENPARENTHESIS_KW	-
array	ID	Symbol Table Entry
[	OPENBRACKET_KW	-
#2	NUMCONST	-
]	CLOSEBRACKET_KW	-
,	COMMA_KW	-
#0	NUMCONST	-
)	CLOSEPARENTHESIS_KW	-
)	CLOSEPARENTHESIS_KW	-
;	SEMICOLON_KW	-
}	CLOSEACCOLADE_KW	-