

# CS 4240 Phase 1

David Zhang

February 23, 2014

## 1 Lexical Rules

Pre-parse the input to remove comments with the following DFA. The resulting NOT-COMMENT tokens are concatenated together.

Everything under the token column represents either generated tokens or custom table actions on the character buffer used to produce the text associated with tokens.

start state	symbol	next state	token
START	$\Sigma - \{/, \}$	START	NOT-COMMENT
START	"	STRING	
START	/	SLASH	
STRING	$\Sigma - \{\\, \}$	STRING	
STRING	\	STRING-SLASH	
STRING	"	START	NOT-COMMENT
STRING-SLASH	$\Sigma$	STRING	
SLASH	$\Sigma - \{*\}$	START	NOT-COMMENT
SLASH	*	COMMENT	
COMMENT	$\Sigma - \{*\}$	COMMENT	
COMMENT	*	COMMENT-END	
COMMENT-END	$\Sigma - \{*, /\}$	COMMENT	
COMMENT-END	*	COMMENT-END	
COMMENT-END	/	START	COMMENT

The DFA for uncommented code.

Note, any time backtracking is mentioned, it essentially is the same as treating the current state as the start state and doing the corresponding transitions or token generations. This is included to simplify the table by removing duplication of the start state transitions.

Note, all ids are later matched character by character with keywords to determine if they are keywords.

Note, drop character for the error state means that the last read character is ignored, and the state remains unchanged.

start state	symbol	next state	token
START	+	START	PLUS
START	-	START	MIN
START	*	START	MULT
START	/	START	DIV
START	=	START	EQ
START	(	START	LPAREN
START	)	START	RPAREN
START	,	START	COMMA

START	&	START	AND
START	—	START	OR
START	[	START	LSQUARE
START	]	START	RSQUARE
START	;	START	SEMI
START	<	LANGLE	
START	>	RANGLE	
START	:	COLON	
START	0-9	INT-LIT	
START	“	STRING-LIT	
START	a-zA-Z	ID	
START	whitespace	START	ignore
START	others	ERROR	drop character
LANGLE	$\Sigma - \{=, \}$	START	LESS , backtrack
LANGLE	>	START	NOTEQ
LANGLE	=	START	LESSEQ
RANGLE	$\Sigma - \{=\}$	START	GREATER, backtrack
RANGLE	=	START	GREATEREQ
COLON	=	START	ASSIGN
COLON	$\Sigma - \{=\}$	START	COLON, backtrack
INT-LIT	0-9	INT-LIT	
INT-LIT	$\Sigma - 0 - 9$	START	INT-LIT, backtrack
ID	a-zA-Z0-9_	ID	
ID	$\Sigma - a - zA - Z0 - 9_$	START	ID, backtrack
STRING-LIT	$\Sigma - \backslash$	STRING-LIT	
STRING-LIT	”	START	STRING-LIT
STRING-LIT	\	STRING-LIT-SLASH	
STRING-LIT-SLASH	n	STRING-LIT	
STRING-LIT-SLASH	t	STRING-LIT	
STRING-LIT-SLASH	”	STRING-LIT	
STRING-LIT-SLASH	\	STRING-LIT	
STRING-LIT-SLASH	^	STRING-LIT-CTL	
STRING-LIT-SLASH	0-9	STRING-LIT-CODE-1	
STRING-LIT-SLASH	whitespace	STRING-LIT-SPACE	ignore 2 characters
STRING-LIT-SLASH	others	ERROR	drop character
STRING-LIT-CTL	@A-Z[\]^_.	STRING-LIT	
STRING-LIT-CTL	others	ERROR	drop character
STRING-LIT-CODE-1	0-9	STRING-LIT-CODE-2	
STRING-LIT-CODE-1	others	ERROR	drop character
STRING-LIT-CODE-2	0-9	STRING-LIT	
STRING-LIT-CODE-2	others	ERROR	drop character
STRING-LIT-SPACE	whitespace	STRING-LIT-SPACE	ignore
STRING-LIT-SPACE	\	STRING-LIT	ignore
STRING-LIT-SPACE	others	ERROR	drop character

## 2 Grammar Rules

Raw Grammar:

symbol	rule
$\langle \text{tiger-program} \rangle$	let $\langle \text{declaration-segment} \rangle$ in $\langle \text{stat-seq} \rangle$ end
$\langle \text{declaration-segment} \rangle$	$\langle \text{type-declaration-list} \rangle \langle \text{var-declaration-list} \rangle \langle \text{funct-declaration-list} \rangle$

$\langle \text{type-declaration-list} \rangle$	$\langle \text{type-declaration} \rangle \langle \text{type-declaration-list} \rangle$
$\langle \text{type-declaration-list} \rangle$	NULL
$\langle \text{var-declaration-list} \rangle$	$\langle \text{var-declaration} \rangle \langle \text{var-declaration-list} \rangle$
$\langle \text{var-declaration-list} \rangle$	NULL
$\langle \text{funct-declaration-list} \rangle$	$\langle \text{funct-declaration} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{funct-declaration-list} \rangle$	NULL
$\langle \text{type-declaration} \rangle$	type id = $\langle \text{type} \rangle$ ;
$\langle \text{type} \rangle$	$\langle \text{type-id} \rangle$
$\langle \text{type} \rangle$	array [ INTLIT ] of $\langle \text{type-dim} \rangle$ of $\langle \text{type-id} \rangle$
$\langle \text{type-dim} \rangle$	[ INTLIT ] $\langle \text{type-dim} \rangle$
$\langle \text{type-dim} \rangle$	NULL
$\langle \text{type-id} \rangle$	int
$\langle \text{type-id} \rangle$	string
$\langle \text{type-id} \rangle$	id
$\langle \text{var-declaration} \rangle$	var $\langle \text{id-list} \rangle$ : $\langle \text{type-id} \rangle$ $\langle \text{optional-init} \rangle$ ;
$\langle \text{id-list} \rangle$	id
$\langle \text{id-list} \rangle$	id , $\langle \text{id-list} \rangle$
$\langle \text{optional-init} \rangle$	NULL
$\langle \text{optional-init} \rangle$	:= $\langle \text{const} \rangle$
$\langle \text{funct-declaration} \rangle$	function id ( $\langle \text{param-list} \rangle$ ) $\langle \text{ret-type} \rangle$ begin $\langle \text{stat-seq} \rangle$ end ;
$\langle \text{param-list} \rangle$	NULL
$\langle \text{param-list} \rangle$	$\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{param-list-tail} \rangle$	NULL
$\langle \text{param-list-tail} \rangle$	, $\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{ret-type} \rangle$	NULL
$\langle \text{ret-type} \rangle$	: $\langle \text{type-id} \rangle$
$\langle \text{param} \rangle$	id : $\langle \text{type-id} \rangle$
$\langle \text{stat-seq} \rangle$	$\langle \text{stat} \rangle \langle \text{stat-seq} \rangle$
$\langle \text{stat-seq} \rangle$	$\langle \text{stat} \rangle$
$\langle \text{stat} \rangle$	$\langle \text{lvalue} \rangle := \langle \text{expr} \rangle$ ;
$\langle \text{stat} \rangle$	if $\langle \text{expr} \rangle$ then $\langle \text{stat-seq} \rangle$ endif ;
$\langle \text{stat} \rangle$	if $\langle \text{expr} \rangle$ then $\langle \text{stat-seq} \rangle$ else $\langle \text{stat-seq} \rangle$ endif ;
$\langle \text{stat} \rangle$	while $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	for id := $\langle \text{expr} \rangle$ to $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	$\langle \text{opt-prefix} \rangle$ id ( $\langle \text{expr-list} \rangle$ ) ;
$\langle \text{stat} \rangle$	break ;
$\langle \text{stat} \rangle$	return $\langle \text{expr} \rangle$ ;
$\langle \text{expr} \rangle$	$\langle \text{expr} \rangle \langle \text{binary-operator} \rangle \langle \text{expr} \rangle$
$\langle \text{expr} \rangle$	$\langle \text{const} \rangle$
$\langle \text{expr} \rangle$	$\langle \text{lvalue} \rangle$
$\langle \text{expr} \rangle$	- $\langle \text{expr} \rangle$
$\langle \text{expr} \rangle$	( $\langle \text{expr} \rangle$ )
$\langle \text{binary-operator} \rangle$	*
$\langle \text{binary-operator} \rangle$	/
$\langle \text{binary-operator} \rangle$	+
$\langle \text{binary-operator} \rangle$	-
$\langle \text{binary-operator} \rangle$	=
$\langle \text{binary-operator} \rangle$	<
$\langle \text{binary-operator} \rangle$	>
$\langle \text{binary-operator} \rangle$	<=
$\langle \text{binary-operator} \rangle$	>=
$\langle \text{binary-operator} \rangle$	<>
$\langle \text{binary-operator} \rangle$	&

$\langle \text{binary-operator} \rangle$	
$\langle \text{binary-operator} \rangle$	:=
$\langle \text{opt-prefix} \rangle$	$\langle \text{lvalue} \rangle :=$
$\langle \text{opt-prefix} \rangle$	NULL
$\langle \text{const} \rangle$	INTLIT
$\langle \text{const} \rangle$	STRLIT
$\langle \text{const} \rangle$	nil
$\langle \text{expr-list} \rangle$	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	NULL
$\langle \text{expr-list-tail} \rangle$	, $\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list-tail} \rangle$	NULL
$\langle \text{lvalue} \rangle$	id $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	[ $\langle \text{expr} \rangle$ ] $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	NULL

Modified Grammar:

symbol	rule
$\langle \text{tiger-program} \rangle$	let $\langle \text{declaration-segment} \rangle$ in $\langle \text{stat-seq} \rangle$ end
$\langle \text{declaration-segment} \rangle$	$\langle \text{type-declaration-list} \rangle \langle \text{var-declaration-list} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{type-declaration-list} \rangle$	$\langle \text{type-declaration} \rangle \langle \text{type-declaration-list} \rangle$
$\langle \text{type-declaration-list} \rangle$	NULL
$\langle \text{var-declaration-list} \rangle$	$\langle \text{var-declaration} \rangle \langle \text{var-declaration-list} \rangle$
$\langle \text{var-declaration-list} \rangle$	NULL
$\langle \text{funct-declaration-list} \rangle$	$\langle \text{funct-declaration} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{funct-declaration-list} \rangle$	NULL
$\langle \text{type-declaration} \rangle$	type id = $\langle \text{type} \rangle$ ;
$\langle \text{var-declaration} \rangle$	var $\langle \text{id-list} \rangle$ : $\langle \text{type-id} \rangle \langle \text{optional-init} \rangle$ ;
$\langle \text{funct-declaration} \rangle$	function id ( $\langle \text{param-list} \rangle$ ) $\langle \text{ret-type} \rangle$ begin $\langle \text{stat-seq} \rangle$ end ;
$\langle \text{type} \rangle$	$\langle \text{type-id} \rangle$
$\langle \text{type} \rangle$	array [ INTLIT ] of $\langle \text{type-dim} \rangle$ of $\langle \text{type-id} \rangle$
$\langle \text{type-dim} \rangle$	[ INTLIT ] $\langle \text{type-dim} \rangle$
$\langle \text{type-dim} \rangle$	NULL
$\langle \text{type-id} \rangle$	id
$\langle \text{id-list} \rangle$	id $\langle \text{id-list-tail} \rangle$
$\langle \text{id-list-tail} \rangle$	, id $\langle \text{id-list-tail} \rangle$
$\langle \text{id-list-tail} \rangle$	NULL
$\langle \text{optional-init} \rangle$	:= $\langle \text{const} \rangle$
$\langle \text{optional-init} \rangle$	NULL
$\langle \text{param-list} \rangle$	$\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{param-list} \rangle$	NULL
$\langle \text{param-list-tail} \rangle$	, $\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{param-list-tail} \rangle$	NULL
$\langle \text{ret-type} \rangle$	: $\langle \text{type-id} \rangle$
$\langle \text{ret-type} \rangle$	NULL
$\langle \text{param} \rangle$	id : $\langle \text{type-id} \rangle$
$\langle \text{stat-seq} \rangle$	$\langle \text{stat} \rangle \langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	$\langle \text{stat} \rangle \langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	NULL
$\langle \text{stat} \rangle$	if $\langle \text{expr} \rangle$ then $\langle \text{stat-seq} \rangle \langle \text{stat-if-tail} \rangle$
$\langle \text{stat} \rangle$	while $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	for id := $\langle \text{expr} \rangle$ to $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	break ;

$\langle \text{stat} \rangle$	return $\langle \text{expr} \rangle$ ;
$\langle \text{stat} \rangle$	id $\langle \text{stat-func-or-assign} \rangle$
$\langle \text{stat-func-or-assign} \rangle$	( $\langle \text{expr-list} \rangle$ ) ;
$\langle \text{stat-func-or-assign} \rangle$	$\langle \text{lvalue-tail} \rangle := \langle \text{stat-assign} \rangle$
$\langle \text{stat-if-tail} \rangle$	else $\langle \text{stat-seq} \rangle$ endif ;
$\langle \text{stat-if-tail} \rangle$	endif ;
$\langle \text{stat-assign} \rangle$	- $\langle \text{unaryminus} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	( $\langle \text{expr} \rangle$ ) $\langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	$\langle \text{const} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	id $\langle \text{stat-assign-id} \rangle$
$\langle \text{stat-assign-id} \rangle$	( $\langle \text{expr-list} \rangle$ ) ;
$\langle \text{stat-assign-id} \rangle$	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	$\langle \text{expr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	$\langle \text{orexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	$\langle \text{compare-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	$\langle \text{term-tail} \rangle$
$\langle \text{expr} \rangle$	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr-tail} \rangle$	$\langle \text{orop} \rangle \langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr-tail} \rangle$	NULL
$\langle \text{orexpr} \rangle$	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr-tail} \rangle$	$\langle \text{andop} \rangle \langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr-tail} \rangle$	NULL
$\langle \text{andexpr} \rangle$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	NULL
$\langle \text{compare} \rangle$	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare-tail} \rangle$	$\langle \text{addop} \rangle \langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare-tail} \rangle$	NULL
$\langle \text{term} \rangle$	$\langle \text{factor} \rangle \langle \text{term-tail} \rangle$
$\langle \text{term-tail} \rangle$	$\langle \text{mulop} \rangle \langle \text{factor} \rangle \langle \text{term-tail} \rangle$
$\langle \text{term-tail} \rangle$	NULL
$\langle \text{factor} \rangle$	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	- $\langle \text{unaryminus} \rangle$
$\langle \text{unaryminus} \rangle$	( $\langle \text{expr} \rangle$ )
$\langle \text{unaryminus} \rangle$	$\langle \text{const} \rangle$
$\langle \text{unaryminus} \rangle$	$\langle \text{lvalue} \rangle$
$\langle \text{const} \rangle$	INTLIT
$\langle \text{const} \rangle$	STRLIT
$\langle \text{const} \rangle$	nil
$\langle \text{orop} \rangle$	
$\langle \text{andop} \rangle$	&
$\langle \text{compop} \rangle$	=
$\langle \text{compop} \rangle$	<>
$\langle \text{compop} \rangle$	>
$\langle \text{compop} \rangle$	<
$\langle \text{compop} \rangle$	>=
$\langle \text{compop} \rangle$	<=
$\langle \text{addop} \rangle$	+
$\langle \text{addop} \rangle$	-
$\langle \text{mulop} \rangle$	*
$\langle \text{mulop} \rangle$	/
$\langle \text{expr-list} \rangle$	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$

$\langle \text{expr-list} \rangle$	NULL
$\langle \text{expr-list-tail} \rangle$	, $\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list-tail} \rangle$	NULL
$\langle \text{lvalue} \rangle$	id $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	[ $\langle \text{expr} \rangle$ ] $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	NULL

non-terminal	first set
$\langle \text{lvalue-tail} \rangle$	[, $\epsilon$
$\langle \text{lvalue} \rangle$	id
$\langle \text{expr-list-tail} \rangle$	,, $\epsilon$
$\langle \text{expr-list} \rangle$	(, nil, STRLIT, INTLIT, id, -, $\epsilon$
$\langle \text{mulop} \rangle$	*, /
$\langle \text{addop} \rangle$	+, -
$\langle \text{compop} \rangle$	=, <, >, <=, >=, <>
$\langle \text{andop} \rangle$	&
$\langle \text{orop} \rangle$	
$\langle \text{const} \rangle$	nil, STRLIT, INTLIT
$\langle \text{unaryminus} \rangle$	(, nil, STRLIT, INTLIT, id
$\langle \text{factor} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{term-tail} \rangle$	*, /, $\epsilon$
$\langle \text{term} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{compare-tail} \rangle$	+, -, $\epsilon$
$\langle \text{compare} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{andexpr-tail} \rangle$	=, <, >, <=, >=, <>, $\epsilon$
$\langle \text{andexpr} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{orexpr} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{orexpr-tail} \rangle$	&, $\epsilon$
$\langle \text{expr} \rangle$	(, nil, STRLIT, INTLIT, id, -
$\langle \text{expr-tail} \rangle$	, $\epsilon$
$\langle \text{stat-assign} \rangle$	id, -, (, nil, STRLIT, INTLIT
$\langle \text{stat-assign-id} \rangle$	[, (, *, /, +, -, =, <, >, <=, >=, <>, &,  , $\epsilon$
$\langle \text{stat-assign-tail} \rangle$	*, /, +, -, =, <, >, <=, >=, <>, &,  , $\epsilon$
$\langle \text{stat-if-tail} \rangle$	else, endif
$\langle \text{stat-func-or-assign} \rangle$	(, :=, [
$\langle \text{stat} \rangle$	if, while, for, break, return, id
$\langle \text{stat-seq} \rangle$	if, while, for, break, return, id
$\langle \text{stat-seq-tail} \rangle$	if, while, for, break, return, id, $\epsilon$
$\langle \text{param} \rangle$	id
$\langle \text{ret-type} \rangle$	:, $\epsilon$
$\langle \text{param-list-tail} \rangle$	,, $\epsilon$
$\langle \text{param-list} \rangle$	id, $\epsilon$
$\langle \text{optional-init} \rangle$	:=, $\epsilon$
$\langle \text{id-list-tail} \rangle$	,, $\epsilon$
$\langle \text{id-list} \rangle$	id
$\langle \text{type-id} \rangle$	id
$\langle \text{type-dim} \rangle$	[, $\epsilon$
$\langle \text{type} \rangle$	array, id
$\langle \text{funct-declaration} \rangle$	function
$\langle \text{var-declaration} \rangle$	var
$\langle \text{type-declaration} \rangle$	type
$\langle \text{funct-declaration-list} \rangle$	function, $\epsilon$
$\langle \text{var-declaration-list} \rangle$	var, $\epsilon$

$\langle \text{type-declaration-list} \rangle$	type, $\epsilon$
$\langle \text{declaration-segment} \rangle$	function, var, type, $\epsilon$
$\langle \text{tiger-program} \rangle$	let

Note, the dollar sign is the end of tokens symbol.

non-terminal	follow set
$\langle \text{lvalue-tail} \rangle$	$:=, *, /, +, -, =, <, >, <=, >=, <>, \&$
$\langle \text{lvalue-tail} \rangle$	$ , \text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}, ), ,, , ], \text{then}, \text{do}, \text{to}, ;$
$\langle \text{expr-list-tail} \rangle$	$)$
$\langle \text{expr-list} \rangle$	$)$
$\langle \text{expr-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}, ), ,, , ], \text{then}, \text{do}, \text{to}, ;$
$\langle \text{orexpr-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end},  , ), ,, , ], \text{then}, \text{do}, \text{to}, ;$
$\langle \text{andexpr-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}, \& , , ), ,, , ], \text{then}, \text{do}, \text{to}, ;$
$\langle \text{compare-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}$
$\langle \text{compare-tail} \rangle$	$\& , , ), ,, , ], \text{then}, \text{do}, \text{to}, ;, =, <, >, <=, >=, <>$
$\langle \text{term-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}, \& , , )$
$\langle \text{term-tail} \rangle$	$, , ], \text{then}, \text{do}, \text{to}, ;, =, <, >, <=, >=, <>, +, -$
$\langle \text{stat-assign-tail} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}$
$\langle \text{stat-assign-id} \rangle$	$\text{if}, \text{while}, \text{for}, \text{break}, \text{return}, \text{id}, \text{endif}, \text{end}$
$\langle \text{stat-seq-tail} \rangle$	$\text{endif}, \text{end}$
$\langle \text{ret-type} \rangle$	$\text{begin}$
$\langle \text{param-list-tail} \rangle$	$)$
$\langle \text{param-list} \rangle$	$)$
$\langle \text{optional-init} \rangle$	$\text{function}, \text{in}$
$\langle \text{id-list-tail} \rangle$	$:$
$\langle \text{type-dim} \rangle$	$\text{of}$
$\langle \text{funct-declaration-list} \rangle$	$\text{in}$
$\langle \text{var-declaration-list} \rangle$	$\text{function}, \text{in}$
$\langle \text{type-declaration-list} \rangle$	$\text{var}, \text{function}, \text{in}$
$\langle \text{declaration-segment} \rangle$	$\text{in}$

Note, if there is no corresponding rule, then that means a parser error is generated.

Note, errors are handled by dropping tokens until a valid token is found.

symbol	next token	rule
$\langle \text{addop} \rangle$	$+$	$+$
$\langle \text{addop} \rangle$	$-$	$-$
$\langle \text{andexpr} \rangle$	$($	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr} \rangle$	$\text{nil}$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr} \rangle$	$\text{STRLIT}$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr} \rangle$	$\text{INTLIT}$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr} \rangle$	$\text{id}$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr} \rangle$	$-$	$\langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$=$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$<$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$>$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$>=$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$<=$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$<>$	$\langle \text{compop} \rangle \langle \text{compare} \rangle \langle \text{andexpr-tail} \rangle$
$\langle \text{andexpr-tail} \rangle$	$\text{if}$	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	$\text{while}$	$\epsilon$

$\langle \text{andexpr-tail} \rangle$	for	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	break	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	return	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	id	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	endif	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	end	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	&	$\epsilon$
$\langle \text{andexpr-tail} \rangle$		$\epsilon$
$\langle \text{andexpr-tail} \rangle$	)	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	,	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	]	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	then	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	do	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	to	$\epsilon$
$\langle \text{andexpr-tail} \rangle$	;	$\epsilon$
$\langle \text{andop} \rangle$	&	&
$\langle \text{compare} \rangle$	(	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare} \rangle$	nil	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare} \rangle$	STRLIT	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare} \rangle$	INTLIT	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare} \rangle$	id	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare} \rangle$	-	$\langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare-tail} \rangle$	+	$\langle \text{addop} \rangle \langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare-tail} \rangle$	-	$\langle \text{addop} \rangle \langle \text{term} \rangle \langle \text{compare-tail} \rangle$
$\langle \text{compare-tail} \rangle$	if	$\epsilon$
$\langle \text{compare-tail} \rangle$	while	$\epsilon$
$\langle \text{compare-tail} \rangle$	for	$\epsilon$
$\langle \text{compare-tail} \rangle$	break	$\epsilon$
$\langle \text{compare-tail} \rangle$	return	$\epsilon$
$\langle \text{compare-tail} \rangle$	id	$\epsilon$
$\langle \text{compare-tail} \rangle$	endif	$\epsilon$
$\langle \text{compare-tail} \rangle$	end	$\epsilon$
$\langle \text{compare-tail} \rangle$	&	$\epsilon$
$\langle \text{compare-tail} \rangle$		$\epsilon$
$\langle \text{compare-tail} \rangle$	)	$\epsilon$
$\langle \text{compare-tail} \rangle$	,	$\epsilon$
$\langle \text{compare-tail} \rangle$	]	$\epsilon$
$\langle \text{compare-tail} \rangle$	then	$\epsilon$
$\langle \text{compare-tail} \rangle$	do	$\epsilon$
$\langle \text{compare-tail} \rangle$	to	$\epsilon$
$\langle \text{compare-tail} \rangle$	;	$\epsilon$
$\langle \text{compare-tail} \rangle$	=	$\epsilon$
$\langle \text{compare-tail} \rangle$	<	$\epsilon$
$\langle \text{compare-tail} \rangle$	>	$\epsilon$
$\langle \text{compare-tail} \rangle$	<=	$\epsilon$
$\langle \text{compare-tail} \rangle$	>=	$\epsilon$
$\langle \text{compare-tail} \rangle$	<>	$\epsilon$
$\langle \text{compop} \rangle$	=	=
$\langle \text{compop} \rangle$	<	<
$\langle \text{compop} \rangle$	>	>
$\langle \text{compop} \rangle$	<=	<=
$\langle \text{compop} \rangle$	>=	>=
$\langle \text{compop} \rangle$	<>	<>



$\langle \text{const} \rangle$	nil	nil
$\langle \text{const} \rangle$	STRLIT	STRLIT
$\langle \text{const} \rangle$	INTLIT	INTLIT
$\langle \text{declaration-segment} \rangle$	function	$\langle \text{type-declaration-list} \rangle \langle \text{var-declaration-list} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{declaration-segment} \rangle$	var	$\langle \text{type-declaration-list} \rangle \langle \text{var-declaration-list} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{declaration-segment} \rangle$	type	$\langle \text{type-declaration-list} \rangle \langle \text{var-declaration-list} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{declaration-segment} \rangle$	in	$\epsilon$
$\langle \text{expr-list} \rangle$	)	$\epsilon$
$\langle \text{expr-list} \rangle$	(	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	nil	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	STRLIT	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	INTLIT	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	id	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list} \rangle$	-	$\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr-list-tail} \rangle$	)	$\epsilon$
$\langle \text{expr-list-tail} \rangle$	,	, $\langle \text{expr} \rangle \langle \text{expr-list-tail} \rangle$
$\langle \text{expr} \rangle$	(	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr} \rangle$	nil	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr} \rangle$	STRLIT	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr} \rangle$	INTLIT	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr} \rangle$	id	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr} \rangle$	-	$\langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr-tail} \rangle$		$\langle \text{orop} \rangle \langle \text{orexpr} \rangle \langle \text{expr-tail} \rangle$
$\langle \text{expr-tail} \rangle$	if	$\epsilon$
$\langle \text{expr-tail} \rangle$	while	$\epsilon$
$\langle \text{expr-tail} \rangle$	for	$\epsilon$
$\langle \text{expr-tail} \rangle$	break	$\epsilon$
$\langle \text{expr-tail} \rangle$	return	$\epsilon$
$\langle \text{expr-tail} \rangle$	id	$\epsilon$
$\langle \text{expr-tail} \rangle$	endif	$\epsilon$
$\langle \text{expr-tail} \rangle$	end	$\epsilon$
$\langle \text{expr-tail} \rangle$	)	$\epsilon$
$\langle \text{expr-tail} \rangle$	,	$\epsilon$
$\langle \text{expr-tail} \rangle$	]	$\epsilon$
$\langle \text{expr-tail} \rangle$	then	$\epsilon$
$\langle \text{expr-tail} \rangle$	do	$\epsilon$
$\langle \text{expr-tail} \rangle$	to	$\epsilon$
$\langle \text{expr-tail} \rangle$	;	$\epsilon$
$\langle \text{factor} \rangle$	(	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	nil	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	STRLIT	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	INTLIT	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	id	$\langle \text{unaryminus} \rangle$
$\langle \text{factor} \rangle$	-	- $\langle \text{unaryminus} \rangle$
$\langle \text{funct-declaration} \rangle$	function	function id ( $\langle \text{param-list} \rangle$ ) $\langle \text{ret-type} \rangle$ begin $\langle \text{stat-seq} \rangle$ end ;
$\langle \text{funct-declaration-list} \rangle$	function	$\langle \text{funct-declaration} \rangle \langle \text{funct-declaration-list} \rangle$
$\langle \text{funct-declaration-list} \rangle$	in	$\epsilon$
$\langle \text{id-list} \rangle$	id	id $\langle \text{id-list-tail} \rangle$
$\langle \text{id-list-tail} \rangle$	:	$\epsilon$
$\langle \text{id-list-tail} \rangle$	,	, id $\langle \text{id-list-tail} \rangle$
$\langle \text{lvalue} \rangle$	id	id $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	[	[ $\langle \text{expr} \rangle$ ] $\langle \text{lvalue-tail} \rangle$
$\langle \text{lvalue-tail} \rangle$	:=	$\epsilon$

$\langle \text{lvalue-tail} \rangle$	*	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	/	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	+	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	-	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	=	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	<	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	>	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	<=	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	>=	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	<>	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	&	$\epsilon$
$\langle \text{lvalue-tail} \rangle$		$\epsilon$
$\langle \text{lvalue-tail} \rangle$	if	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	while	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	for	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	break	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	return	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	id	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	endif	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	end	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	)	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	,	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	]	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	then	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	do	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	to	$\epsilon$
$\langle \text{lvalue-tail} \rangle$	;	$\epsilon$
$\langle \text{mulop} \rangle$	*	*
$\langle \text{mulop} \rangle$	/	/
$\langle \text{optional-init} \rangle$	:=	:= $\langle \text{const} \rangle$
$\langle \text{optional-init} \rangle$	function	$\epsilon$
$\langle \text{optional-init} \rangle$	in	$\epsilon$
$\langle \text{orexpr} \rangle$	(	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr} \rangle$	nil	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr} \rangle$	STRLIT	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr} \rangle$	INTLIT	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr} \rangle$	id	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr} \rangle$	-	$\langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr-tail} \rangle$	&	$\langle \text{andop} \rangle \langle \text{andexpr} \rangle \langle \text{orexpr-tail} \rangle$
$\langle \text{orexpr-tail} \rangle$	if	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	while	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	for	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	break	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	return	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	id	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	endif	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	end	$\epsilon$
$\langle \text{orexpr-tail} \rangle$		$\epsilon$
$\langle \text{orexpr-tail} \rangle$	)	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	,	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	]	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	then	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	do	$\epsilon$

$\langle \text{orexpr-tail} \rangle$	to	$\epsilon$
$\langle \text{orexpr-tail} \rangle$	;	$\epsilon$
$\langle \text{orop} \rangle$		
$\langle \text{param} \rangle$	id	id : $\langle \text{type-id} \rangle$
$\langle \text{param-list} \rangle$	)	$\epsilon$
$\langle \text{param-list} \rangle$	id	$\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{param-list-tail} \rangle$	)	$\epsilon$
$\langle \text{param-list-tail} \rangle$	,	, $\langle \text{param} \rangle \langle \text{param-list-tail} \rangle$
$\langle \text{ret-type} \rangle$	begin	$\epsilon$
$\langle \text{ret-type} \rangle$	:	: $\langle \text{type-id} \rangle$
$\langle \text{stat-assign} \rangle$	id	id $\langle \text{stat-assign-id} \rangle$
$\langle \text{stat-assign} \rangle$	-	- $\langle \text{unaryminus} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	(	( $\langle \text{expr} \rangle$ ) $\langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	nil	$\langle \text{const} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	STRLIT	$\langle \text{const} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign} \rangle$	INTLIT	$\langle \text{const} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	end	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	endif	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	id	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	return	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	break	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	for	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	while	$\epsilon$
$\langle \text{stat-assign-id} \rangle$	if	$\epsilon$
$\langle \text{stat-assign-id} \rangle$		$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	&	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	<>	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	>=	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	<=	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	>	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	<	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	=	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	-	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	+	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	/	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	*	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-id} \rangle$	(	( $\langle \text{expr-list} \rangle$ ) ;
$\langle \text{stat-assign-id} \rangle$	[	$\langle \text{lvalue-tail} \rangle \langle \text{stat-assign-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	end	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	endif	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	id	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	return	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	break	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	for	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	while	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$	if	$\epsilon$
$\langle \text{stat-assign-tail} \rangle$		$\langle \text{expr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	&	$\langle \text{orexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	<>	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	>=	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	<=	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	>	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	<	$\langle \text{andexpr-tail} \rangle$

$\langle \text{stat-assign-tail} \rangle$	=	$\langle \text{andexpr-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	-	$\langle \text{compare-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	+	$\langle \text{compare-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	/	$\langle \text{term-tail} \rangle$
$\langle \text{stat-assign-tail} \rangle$	*	$\langle \text{term-tail} \rangle$
$\langle \text{stat-func-or-assign} \rangle$	(	( $\langle \text{expr-list} \rangle$ ) ;
$\langle \text{stat-func-or-assign} \rangle$	:=	$\langle \text{lvalue-tail} \rangle := \langle \text{stat-assign} \rangle$
$\langle \text{stat-func-or-assign} \rangle$	[	$\langle \text{lvalue-tail} \rangle := \langle \text{stat-assign} \rangle$
$\langle \text{stat-if-tail} \rangle$	else	else $\langle \text{stat-seq} \rangle$ endif ;
$\langle \text{stat-if-tail} \rangle$	endif	endif ;
$\langle \text{stat} \rangle$	if	if $\langle \text{expr} \rangle$ then $\langle \text{stat-seq} \rangle$ $\langle \text{stat-if-tail} \rangle$
$\langle \text{stat} \rangle$	while	while $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	for	for id := $\langle \text{expr} \rangle$ to $\langle \text{expr} \rangle$ do $\langle \text{stat-seq} \rangle$ enddo ;
$\langle \text{stat} \rangle$	break	break ;
$\langle \text{stat} \rangle$	return	return $\langle \text{expr} \rangle$ ;
$\langle \text{stat} \rangle$	id	id $\langle \text{stat-func-or-assign} \rangle$
$\langle \text{stat-seq} \rangle$	if	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq} \rangle$	while	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq} \rangle$	for	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq} \rangle$	break	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq} \rangle$	return	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq} \rangle$	id	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	endif	$\epsilon$
$\langle \text{stat-seq-tail} \rangle$	end	$\epsilon$
$\langle \text{stat-seq-tail} \rangle$	if	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	while	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	for	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	break	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	return	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{stat-seq-tail} \rangle$	id	$\langle \text{stat} \rangle$ $\langle \text{stat-seq-tail} \rangle$
$\langle \text{term} \rangle$	-	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term} \rangle$	id	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term} \rangle$	INTLIT	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term} \rangle$	STRLIT	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term} \rangle$	nil	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term} \rangle$	(	$\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term-tail} \rangle$	*	$\langle \text{mulop} \rangle$ $\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term-tail} \rangle$	/	$\langle \text{mulop} \rangle$ $\langle \text{factor} \rangle$ $\langle \text{term-tail} \rangle$
$\langle \text{term-tail} \rangle$	)	$\epsilon$
$\langle \text{term-tail} \rangle$		$\epsilon$
$\langle \text{term-tail} \rangle$	&	$\epsilon$
$\langle \text{term-tail} \rangle$	end	$\epsilon$
$\langle \text{term-tail} \rangle$	endif	$\epsilon$
$\langle \text{term-tail} \rangle$	id	$\epsilon$
$\langle \text{term-tail} \rangle$	return	$\epsilon$
$\langle \text{term-tail} \rangle$	break	$\epsilon$
$\langle \text{term-tail} \rangle$	for	$\epsilon$
$\langle \text{term-tail} \rangle$	while	$\epsilon$
$\langle \text{term-tail} \rangle$	if	$\epsilon$
$\langle \text{term-tail} \rangle$	-	$\epsilon$
$\langle \text{term-tail} \rangle$	+	$\epsilon$
$\langle \text{term-tail} \rangle$	<>	$\epsilon$
$\langle \text{term-tail} \rangle$	>=	$\epsilon$

$\langle \text{term-tail} \rangle$	$\leq$	$\epsilon$
$\langle \text{term-tail} \rangle$	$>$	$\epsilon$
$\langle \text{term-tail} \rangle$	$<$	$\epsilon$
$\langle \text{term-tail} \rangle$	$=$	$\epsilon$
$\langle \text{term-tail} \rangle$	$;$	$\epsilon$
$\langle \text{term-tail} \rangle$	to	$\epsilon$
$\langle \text{term-tail} \rangle$	do	$\epsilon$
$\langle \text{term-tail} \rangle$	then	$\epsilon$
$\langle \text{term-tail} \rangle$	$]$	$\epsilon$
$\langle \text{term-tail} \rangle$	$,$	$\epsilon$
$\langle \text{tiger-program} \rangle$	let	let $\langle \text{declaration-segment} \rangle$ in $\langle \text{stat-seq} \rangle$ end
$\langle \text{type} \rangle$	array	array [ INTLIT ] of $\langle \text{type-dim} \rangle$ of $\langle \text{type-id} \rangle$
$\langle \text{type} \rangle$	id	$\langle \text{type-id} \rangle$
$\langle \text{type-declaration-list} \rangle$	type	$\langle \text{type-declaration} \rangle \langle \text{type-declaration-list} \rangle$
$\langle \text{type-declaration-list} \rangle$	var	$\epsilon$
$\langle \text{type-declaration-list} \rangle$	function	$\epsilon$
$\langle \text{type-declaration-list} \rangle$	in	$\epsilon$
$\langle \text{type-declaration} \rangle$	type	type id = $\langle \text{type} \rangle$ ;
$\langle \text{type-dim} \rangle$	[	[ INTLIT ] $\langle \text{type-dim} \rangle$
$\langle \text{type-dim} \rangle$	of	$\epsilon$
$\langle \text{type-id} \rangle$	id	id
$\langle \text{unaryminus} \rangle$	(	( $\langle \text{expr} \rangle$ )
$\langle \text{unaryminus} \rangle$	nil	$\langle \text{const} \rangle$
$\langle \text{unaryminus} \rangle$	STRLIT	$\langle \text{const} \rangle$
$\langle \text{unaryminus} \rangle$	INTLIT	$\langle \text{const} \rangle$
$\langle \text{unaryminus} \rangle$	id	$\langle \text{lvalue} \rangle$
$\langle \text{var-declaration-list} \rangle$	function	$\epsilon$
$\langle \text{var-declaration-list} \rangle$	in	$\epsilon$
$\langle \text{var-declaration-list} \rangle$	var	$\langle \text{var-declaration} \rangle \langle \text{var-declaration-list} \rangle$
$\langle \text{var-declaration} \rangle$	var	var $\langle \text{id-list} \rangle$ : $\langle \text{type-id} \rangle \langle \text{optional-init} \rangle$ ;