```
[1]
     <PROGRAM>
                             \rightarrow <IDENT><ENV><DATA><PROC>
[2]
     <IDENT>
                                identification division <END INST> program-id.
                                 ID<END INST> author. WORDS<END INST>
                                 date-written. WORDS <END INST>
     <WORDS>
[3]
                                 ID <WORDS LR>
[4]
     <WORDS LR>
                                ID <WORDS LR>
[5]
[6]
     <END INST>
                                \cdot n
                                 environment division
<br/>
\langle {\rm END\_INST} \rangle configuration
[7]
     <ENV>
                                 section<END_INST> source-computer.
                                 WORDS<END INST> object-computer.
                                 WORDS<END INST>
                                 data division<END INST> working-storage
[8]
     <DATA>
                                 section<END INST> <VAR LIST>
[9]
     <VAR LIST>
                                 <VAR DECL> <VAR LIST>
[10]
     <VAR DECL>
                             \rightarrow <LEVEL> ID pic IMAGE <VAR DECL TAIL>
[11]
[12]
    <VAR_DECL_TAIL>
                                value INTEGER<END INST>
[13]
                                 <END INST>
[14]
     <LEVEL>
                                INTEGER
     <PROC>
                                 procedure division<END INST> ID
[15]
                                 section<END INST> <LABELS> end program ID.
                                 <LABEL><END_INST> <INSTRUCTION_LIST>
[16]
    <LABELS>
                                 <LABELS LR>
[17]
                                 <LABEL><END INST> <INSTRUCTION LIST>
     <LABELS LR>
                                 <LABELS LR>
[18]
                                \varepsilon
[19]
     <LABEL>
                                ID
[20]
     <INSTRUCTION_LIST>
                                <INSTRUCTION> <INSTRUCTION LIST>
[21]
[22]
     <INSTRUCTION>
                                <ASSIGNATION>
[23]
                             \rightarrow <IF>
[24]
                             \rightarrow <CALL>
[25]
                             \rightarrow <READ>
[26]
                                <WRITE>
[27]
                                stop run<END INST>
[28]
    <ASSIGNATION>
                                 move <EXPRESSION> to ID<END INST>
[29]
                                 compute ID = <EXPRESSION><END INST>
[30]
                                 add <EXPRESSION> to ID<END INST>
                                 subtract < EXPRESSION > from \ ID < END \quad INST >
[31]
[32]
                                multiply <ASSIGN END><END INST>
[33]
                                 divide <ASSIGN_END><END_INST>
[34]
    <ASSIGN END>
                                 <EXPRESSION>,<EXPRESSION> giving ID
[35]
     <EXPRESSION>

ightarrow <EXP AND> <EXPRESSION LR>
[36]
    <EXPRESSION LR>
                             \rightarrow or <EXP AND> <EXPRESSION LR>
[37]
                             \rightarrow
    \langle \text{EXP\_AND} \rangle

ightarrow <EXP EQUAL> <EXP AND LR>
[38]
[39]
    <EXP AND LR>
                                 and <EXP EQUAL> <EXP AND LR>
```

```
[40]
[41]
    <EXP EQUAL>
                            \rightarrow <EXP ADD> <EXP EQUAL LR>
[42]
    <EXP EQUAL LR>
                            \rightarrow = <EXP ADD> <EXP EQUAL LR>
[43]

ightarrow < <EXP ADD> <EXP EQUAL LR>
                            \rightarrow > <EXP ADD> <EXP EQUAL LR>
[44]
[45]
                               <= <EXP_ADD> <EXP_EQUAL_LR>
                               >= <EXP ADD> <EXP EQUAL LR>
[46]
[47]
    <EXP ADD>
                               <EXP MULT> <EXP ADD LR>
[48]
    <EXP\_ADD\_LR>
                            \rightarrow + <EXP MULT> <EXP ADD LR>
[49]
[50]
                            \rightarrow - <EXP MULT> <EXP ADD LR>
[51]
[52]
    <EXP_NOT> <EXP_MULT_LR>
                            [53]
    <EXP_MULT_LR>
                               / < EXP NOT > < EXP MULT LR >
[54]
                            \rightarrow
[55]
[56]
    <EXP_NOT>
                            \rightarrow -<EXP NOT>
[57]
                               not <EXP_NOT>
[58]
                               <EXP PARENTHESIS>
[59]
    <EXP PARENTHESIS>
                            \rightarrow (\langle EXPRESSION \rangle)
[60]

ightarrow <EXP TERM>
    [61]
                            \rightarrow ID
[62]
                               INTEGER
                            \rightarrow
[63]
                            \rightarrow true
[64]
                               false
[65]
    \langle IF \rangle
                               if <EXPRESSION> then <INSTRUCTION_LIST>
                                <IF END>
[66]
    <IF END>
                                else <INSTRUCTION LIST> end-if
[67]
                                end-if
[68]
    <CALL>
                                perform ID < CALL_TAIL>
                                until <EXPRESSION><END INST>
[69]
    <CALL_TAIL>
                               <END INST>
[70]
[71]
    <READ>
                                accept ID<END INST>
[72]
                                display <WRITE TAIL>
    <WRITE>
[73]
    <WRITE_TAIL>
                               <EXPRESSION><END_INST>
[74]
                                STRING<END INST>
```

Variable	First^1	$Follow^1$
<program></program>	identification	
<IDENT $>$	identification	environment
<WORDS $>$	ID	
$<$ WORDS_LR $>$	ID, ε	

<end_inst></end_inst>		program-id, date-written, environment, configuration, source-computer, object-computer, data, working-storage, INTEGER, ε , ID, move, compute, add, substract, multiply, divide, if, perform, accept, display, stop
<env></env>	environment	data
<data></data>	data	procedure
<var list=""></var>	INTEGER, ε	-
<var_list></var_list>	INTEGER, E INTEGER	procedure
_		INTEGER, ε
<var_decl_tail></var_decl_tail>	value, .	INTEGER, ε
<level></level>	INTEGER	ID
<proc></proc>	procedure	1
<labels></labels>	ID	end
<labels_lr></labels_lr>	ID, ε	end
<label></label>	ID	·
<instruction_list></instruction_list>	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε	ID, ε
<instruction></instruction>	move, compute, add, substract,	move, compute, add, substract, mul-
	multiply, divide, if, perform,	tiply, divide, if, perform, accept, dis-
	accept, display, stop	play, stop, ε
<assignation></assignation>	move, compute, add, substract,	move, compute, add, substract, mul-
(118818111111111)	multiply, divide	tiply, divide, if, perform, accept, dis-
	maroipi, arvide	play, stop, ε
<assign end=""></assign>	-, not, (, ID, INTEGER, true, false	piaj, stop, c
<expression></expression>	-, not, (, ID, INTEGER, true, false	to, ., from, , , giving,), then
<expression lr=""></expression>	or, ε	to, ., from, , , giving,), then
<exp and=""></exp>	-, not, (, ID, INTEGER, true, false	or, ε
<exp and="" lr=""></exp>	and, ε	or, ε
<exp equal=""></exp>	-, not, (, ID, INTEGER, true, false	and, ε
<u> </u>		_^
<exp_equal_lr> <exp_add></exp_add></exp_equal_lr>	$=,<,>,<=,>=,\varepsilon$ -, not, (, ID, INTEGER, true, false	and, ε
<exp_add_lr></exp_add_lr>	$+$, $-$, ε	
<exp mult=""></exp>		$=,<,>,<=,>=,\varepsilon$
<u> </u>	-, not, (, ID, INTEGER, true, false	$+$, -, ε
<exp_mult_lr></exp_mult_lr>	*, $/$, ε	+, -, ε * / -
<exp_not></exp_not>	-, not, (, ID, INTEGER, true, false	*, /, ε
<exp_parenthesis></exp_parenthesis>	(, ID, INTEGER, true, false	*, /, ε
<exp_term></exp_term>	ID, INTEGER, true, false	*,/,arepsilon
<if></if>	if	move, compute, add, substract, mul-
		tiply, divide, if, perform, accept, dis-
ID DND	1 1.0	play, stop, ε
<if_end></if_end>	else, end-if	move, compute, add, substract, mul-
		tiply, divide, if, perform, accept, dis-
		play, stop, ε

<call></call>	perform	move, compute, add, substract, multiply, divide, if, perform, accept, dis-
$<$ CALL_TAIL $>$	until, .	play, stop, ε move, compute, add, substract, multiply, divide, if, perform, accept, display stop, ε
<read></read>	accept	play, stop, ε move, compute, add, substract, multiply, divide, if, perform, accept, dis-
<write></write>	display	play, stop, ε move, compute, add, substract, multiply, divide, if, perform, accept, dis-
$<$ WRITE_TAIL $>$	STRING, -, not, (, ID, INTEGER, true, false	play, stop, ε move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε

PROGRAM> 1 JIDENT> 2 WORDS> 3 WORDS_LR> 4 5 END_INST> 6 ENV> 7 DATA> 8 VAR_LIST> 9 10 VAR_DECL> 11 VAR_DECL_TAIL> 13 12 LEVEL> 14 PROC> 14 LABELS 16 LABELS_LR> 17 LABEL> 19 INSTRUCTION_LIST> 21 INSTRUCTION> 34 34 ASSIGNATION> 35 35 EXPRESSION LR> 37 37 EXP_AND_LR> 41 41 EXP_EQUAL LR> 41 41 EXP_EQUAL_LR> 48 48 EXP_ADD LR> 52 52 EXP_MULT LR> 52 52 EXP_NOT> 56 56 EXP_NOT> 56 56		identification	ID		environment	data	INTEGER	value
WORDS LR > 4 5 <end_inst> 6 <env> 7 <data> 8 <var_list> 9 10 <var_decl> 11 11 <var_decl_tail> 13 12 <level> 14 4 <proc> 14 4 <labels lr=""> 17 4 <label> 19 4 <instruction list=""> 21 4 <instruction> 34 34 <assignation> 35 35 <assignend> 35 35 <expression lr=""> 37 38 <exp_and> 38 38 <exp_and lr=""> 41 41 <exp_equal lr=""> 48 48 <exp_add lr=""> 52 52 <exp_mult> 52 52 <exp_mult lr=""> 52 52 <exp_mult lr=""> 56 56</exp_mult></exp_mult></exp_mult></exp_add></exp_equal></exp_and></exp_and></expression></assignend></assignation></instruction></instruction></label></labels></proc></level></var_decl_tail></var_decl></var_list></data></env></end_inst>	<PROGRAM $>$	1						
<words_lr> 4 5 <end_inst> 6 <env> 7 <data> 8 <var_list> 9 10 <var_decl> 11 11 <var_decl_tail> 13 12 <level> 14 4 <proc> 14 4 <labels_lr> 16 4 <labels_lr> 17 4 <instruction_list> 21 4 <instruction> 34 34 <assignation> 35 35 <assignend> 34 34 <expression_lr> 37 35 <expand_lr> 38 38 <exp_and_lr> 41 41 <exp_equal_lr> 41 41 <exp_add_lr> 48 48 <exp_mult< td=""> 52 52 <exp_mult_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_mult_lr></exp_mult<></exp_add_lr></exp_equal_lr></exp_and_lr></expand_lr></expression_lr></assignend></assignation></instruction></instruction_list></labels_lr></labels_lr></proc></level></var_decl_tail></var_decl></var_list></data></env></end_inst></words_lr>	<IDENT $>$	2						
<end_inst> 6 <env> 7 <data> 8 <var_list> 9 10 <var_decl> 11 <var_decl_tail> 13 12 <level> 14 <proc> 14 <labels> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 34 <assignation> 35 35 <expression_lr> 37 52 <exp_and_lr> 41 41 <exp_add_lr> 48 48 <exp_add_lr> 48 48 <exp_add_lr> 52 52 <exp_mult_l< td=""> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_mult_l<></exp_add_lr></exp_add_lr></exp_add_lr></exp_and_lr></expression_lr></assignation></instruction></instruction_list></label></labels_lr></labels></proc></level></var_decl_tail></var_decl></var_list></data></env></end_inst>	<WORDS $>$		3					
ENV> 7 DATA> 8 <var_list> 9 10 VAR_DECL> 11 11 <var_decl_tail> 13 12 <level> 14 14 <proc> 14 14 <labels> 16 17 18 <label> 19 19 19 19 19 10 <instruction_list> 21 11 11 11 11 11 11 11 11 11 11 11 12</instruction_list></label></labels></proc></level></var_decl_tail></var_list>	$<$ WORDS_LR $>$		4	5				
<data> 8 <var_list> 9 10 <var_decl> 11 11 <var_decl_tail> 13 12 <level> 14 14 <proc> 14 14 <labels< td=""> 16 15 <labels_lr> 17 17 <label> 19 18 <instruction_list> 21 18 <assignation> 34 34 <assign_end> 34 34 <expression_lr> 37 35 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_equal_lr> 41 41 <exp_add_lr> 48 48 <exp_add_lr> 52 52 <exp_mult> 52 52 <exp_mult_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_mult_lr></exp_mult></exp_add_lr></exp_add_lr></exp_equal_lr></exp_and_lr></exp_and></expression_lr></assign_end></assignation></instruction_list></label></labels_lr></labels<></proc></level></var_decl_tail></var_decl></var_list></data>	$<\!\!\operatorname{END_INST}\!\!>$			6				
VAR_LIST> 9 10 VAR_DECL> 11 12 VAR_DECL_TAIL> 13 12 VEVEL> 14 14 PROC> 16 14 14 CLABELS> 16 16 16 16 16 CLABEL> 19 19 19 19 10 11 12 12 12 10 11 12 12 12 10 10 10 10 11 12 </td <td><ENV$>$</td> <td></td> <td></td> <td></td> <td>7</td> <td></td> <td></td> <td></td>	<ENV $>$				7			
<var_decl> 11 <var_decl_tail> 13 12 <level> 14 <proc> 16 17 <labels_lr> 17 18 <label> 19 19 <instruction_list> 21 19 <instruction> 34 34 <assign_end> 34 34 <expression> 35 35 <expression_lr> 37 22 <exp_and< td=""> 38 38 <exp_and_lr> 41 41 <exp_equal> 41 41 <exp_equal_lr> 48 48 <exp_add_lr> 52 52 <exp_mult< td=""> 52 52 <exp_mult_lr> 52 52 <exp_not> 56 56</exp_not></exp_mult_lr></exp_mult<></exp_add_lr></exp_equal_lr></exp_equal></exp_and_lr></exp_and<></expression_lr></expression></assign_end></instruction></instruction_list></label></labels_lr></proc></level></var_decl_tail></var_decl>	<DATA $>$					8		
VAR_DECL_TAIL> 13 12 <level> 14 <proc> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assignation> 35 <assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and< td=""> 38 <exp_and_lr> 41 <exp_equal< td=""> 41 <exp_equal_lr> 48 <exp_add< td=""> 48 <exp_add_lr> 52 <exp_mult< td=""> 52 <exp_mult_lr> 52 <exp_mult_lr> 52 <exp_not> 56</exp_not></exp_mult_lr></exp_mult_lr></exp_mult<></exp_add_lr></exp_add<></exp_equal_lr></exp_equal<></exp_and_lr></exp_and<></expression_lr></expression></assign_end></assignation></instruction></instruction_list></label></labels_lr></proc></level>	$<$ VAR_LIST $>$					9		10
<level> 14 <proc> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assignation> 35 <assign_end> 35 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 41 <exp_add> 48 48 <exp_add_lr> 52 52 <exp_mult> 52 52 <exp_mult_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end></assignation></instruction></instruction_list></label></labels_lr></proc></level>	_					11		
<proc> LABELS> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add_lr> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 52 <exp_mult_lr> 52 <exp_mult_lr> 55 <exp_not> 56</exp_not></exp_mult_lr></exp_mult_lr></exp_mult_lr></exp_mult></exp_add_lr></exp_add_lr></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end></instruction></instruction_list></label></labels_lr></proc>	$<$ VAR_DECL_TAIL $>$			13				12
<labels> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add lr=""> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 52 <exp_mult_lr> 55 <exp_not> 56</exp_not></exp_mult_lr></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end></instruction></instruction_list></label></labels_lr></labels>	$<\!\!\mathrm{LEVEL}\!\!>$						14	
<labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assignation> 34 <assign_end> 35 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal< td=""> 41 <exp_add> 48 <exp_add_lr> 48 <exp_mult> 52 <exp_mult_lr> 52 <exp_mult_lr> 56</exp_mult_lr></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal<></exp_and_lr></exp_and></expression_lr></expression></assign_end></assignation></instruction></instruction_list></label></labels_lr>	<proc></proc>							
<label> 19 <instruction_list> 21 <instruction> 34 <assignation> 34 <assign_end> 35 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add< td=""> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 52 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_add<></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end></assignation></instruction></instruction_list></label>	<LABELS $>$		16					
<instruction_list> 21 <instruction> 34 <assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add< td=""> 48 <exp_add_lr> 52 <exp_mult< td=""> 52 <exp_mult_lr> 52 <exp_not> 56</exp_not></exp_mult_lr></exp_mult<></exp_add_lr></exp_add<></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end></instruction></instruction_list>	$<$ LABELS_LR $>$		17					
<instruction> <assignation> 34 34 <expression> 35 35 <expression_lr> 37 38 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_equal> 41 41 <exp_equal_lr> 48 48 <exp_add> 48 48 <exp_mult> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_mult></exp_add></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assignation></instruction>			19					
<assignation> <assign_end> 34 34 <expression> 35 35 <exp_ession_lr> 37 38 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_equal> 41 41 <exp_add> 48 48 <exp_add_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_add_lr></exp_add></exp_equal></exp_and_lr></exp_and></exp_ession_lr></expression></assign_end></assignation>	$<$ INSTRUCTION_LIST $>$		21					
<assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add_lr> 48 <exp_mult> 52 <exp_mult_lr> 52 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression></assign_end>	<INSTRUCTION $>$							
<expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add< td=""> 48 <exp_add_lr> 52 <exp_mult< td=""> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult<></exp_add_lr></exp_add<></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></expression>								
<expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_add> 48 <exp_add_lr> 48 <exp_mult> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal></exp_and_lr></exp_and></expression_lr>	_		34					
<exp_and> 38 <exp_and_lr> 41 <exp_equal> 41 <exp_equal_lr> 48 <exp_add> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal_lr></exp_equal></exp_and_lr></exp_and>			35				35	
<exp_and_lr> <exp_equal> 41 41 <exp_equal_lr> 48 48 <exp_add_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_add_lr></exp_equal_lr></exp_equal></exp_and_lr>	_			37				
<exp_equal> 41 41 <exp_equal_lr> 48 48 <exp_add_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_add_lr></exp_equal_lr></exp_equal>	_		38				38	
<exp_equal_lr> <exp_add> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal_lr>								
<exp_add> 48 48 <exp_add_lr> 52 52 <exp_mult_lr> 56 56</exp_mult_lr></exp_add_lr></exp_add>			41				41	
<exp_add_lr> <exp_mult> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult></exp_add_lr>								
<exp_mult> 52 <exp_mult_lr> 56 <exp_not> 56</exp_not></exp_mult_lr></exp_mult>	_		48				48	
<exp_mult_lr> <exp_not> 56</exp_not></exp_mult_lr>								
<exp_not> 56</exp_not>	_		52				52	
_								
\sim EYP PARENTHESIS \sim 150 50	_							
CEAT_TAILENTHEDIS/ 59	<exp_parenthesis></exp_parenthesis>		59				59	

<exp_term> <if></if></exp_term>		61				62	
$<$ IF $_{ m END}>$							
<call></call>							
<call_tail></call_tail>							
<read> <write></write></read>							
<write_tail></write_tail>		73				73	
	II						
	procedure	morro	compute	add	substract	multiply	divide
<program></program>	procedure	move	compute	auu	Substract	шиппріу	arvide
<ident></ident>							
<words></words>							
<WORDS LR $>$							
$<\!\!\operatorname{END_INST}\!\!>$							
<ENV $>$							
<data></data>							
$<$ VAR_LIST $>$	10						
$<$ VAR_DECL $>$							
$<$ VAR_DECL_TAIL $>$							
<LEVEL $>$							
<proc></proc>	15						
<labels></labels>							
<labels_lr></labels_lr>							
<label></label>		20	20	20	20	20	20
<instruction_list></instruction_list>		20	20	20	20	20	20
<instruction></instruction>		22	22	22	22	22	22
<assignation></assignation>		28	29	30	31	32	33
<assign_end> <expression></expression></assign_end>							
<expression lr=""></expression>							
<exp and=""></exp>							
<exp and="" lr=""></exp>							
<exp equal=""></exp>							
<exp equal="" lr=""></exp>							
$\langle ext{EXP_ADD} angle$							
$<$ EXP $_$ ADD $_$ LR $>$							
$<$ EXP $_$ MULT $>$							
$<$ EXP $_$ MULT $_$ LR $>$							
$\langle \text{EXP_NOT} \rangle$							
<exp_parenthesis></exp_parenthesis>							
<exp_term></exp_term>							
<if></if>							
<if_end></if_end>							
<call></call>							
<call_tail></call_tail>							
<read></read>							
<write></write>	II						

	perform	accept	display	stop	_	not	(true	false
<program></program>	1	•		•					
<IDENT $>$									
<words></words>									
<words lr=""></words>									
<end inst=""></end>									
<env></env>									
<data></data>									
<var_list></var_list>									
									
<var_decl></var_decl>									
<var_decl_tail></var_decl_tail>									
<LEVEL $>$									
<proc></proc>									
<LABELS $>$									
$<$ LABELS_LR $>$									
<L $ABEL>$									
$<$ INSTRUCTION_LIST $>$	20	20	20	20					
<INSTRUCTION $>$	24	25	26	27					
<ASSIGNATION $>$									
<ASSIGN END $>$					34	34	34	34	34
<expression></expression>					35	35	35	35	35
<expression lr=""></expression>									
$<$ EXP AND $>$ $^-$					38	38	38	38	38
<exp and="" lr=""></exp>									
<exp_equal></exp_equal>					41	41	41	41	41
<exp_equal_lr></exp_equal_lr>									
<exp add=""></exp>					48	48	48	48	48
<exp add="" lr=""></exp>					50	10	10	10	10
<exp_mult></exp_mult>					52	52	52	52	52
<exp lr="" mult=""></exp>					55	52	02	32	52
<exp_not></exp_not>					56	57	58	58	58
_					50	51			
<exp_parenthesis></exp_parenthesis>							59	60	60
<exp_term></exp_term>								63	64
<if></if>									
<if_end></if_end>									
<call></call>	68								
$<$ CALL $_{-}$ TAIL $>$									
<READ $>$		71							
<WRITE $>$			72						
$<$ WRITE_TAIL $>$					73	73	73	73	73