```
[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>
                                 ID < WORDS \quad LR >
     <WORDS>
[3]
[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>
[8]
     <DATA>
                                 data division<END INST> working-storage
                                 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>
                             \rightarrow
[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>
                             \rightarrow <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>
[43]
                                  \rightarrow < <EXP ADD>
                                  \rightarrow > < EXP_ADD>
[44]
[45]
                                  \rightarrow <= <EXP_ADD>
                                  \rightarrow >= <EXP ADD>
[46]
[47]
[48]
     <EXP ADD>

ightarrow <EXP MULT> <EXP ADD LR>
     <EXP\_ADD\_LR>
                                  \rightarrow + <EXP MULT> <EXP ADD LR>
[49]
[50]
                                  \rightarrow - <EXP MULT> <EXP ADD LR>
[51]
[52]
     \rightarrow <EXP_NOT> <EXP_MULT_LR>
                                  \rightarrow \quad *<\! \texttt{EXP} \quad \texttt{NOT}\! ><\! \texttt{EXP} \quad \texttt{MULT} \quad \texttt{LR}\! >
[53]
     <EXP_MULT_LR>
                                  \rightarrow / <EXP NOT> <EXP MULT LR>
[54]
[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]
     <\!\!\mathrm{EXP\_TERM}\!\!>
                                  \rightarrow ID
[62]
                                     INTEGER
                                  \rightarrow
[63]
                                  \rightarrow true
[64]
                                  \rightarrow false
[65]
     \langle IF \rangle
                                  \rightarrow 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>
[70]
                                      <END INST>
[71]
     <READ>
                                      accept ID<END INST>
[72]
     <WRITE>
                                      display <WRITE TAIL>
[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>perform move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε <CALL TAIL>until, . move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε <READ> move, compute, add, substract, mulaccept tiply, divide, if, perform, accept, display, stop, ε <WRITE>display move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε <WRITE_TAIL>STRING, -, not, (, ID, INTEGER, move, compute, add, substract, multrue, false tiply, divide, if, perform, accept, display, stop, ε

PROGRAM> 1 IDENT> 2 WORDS> 3 <words lr=""> 4 5 &END_INST> 6 CENV> 7 DATA> 8 <var_list> 9 10 <var_decl> 11 12 <uran_decl_tail> 13 12 <uran_decl_tail> 14 4 <uran_decl_tail> 13 12 <uran_decl_tail> 14 4 <uran_decl_tail> 14 4 <uran_decl_tail> 13 12 <uran_decl_tail> 14 4 <uran_decl_tail> 14 4 <uran_decl_tail> 15 16 <uran_decl_tail> 31 12 <uran_decl_tail> 31 34 34 <uran_decl_tail> 31 34 34 34 3</uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></uran_decl_tail></var_decl></var_list></words>		identification	ID		environment	data	INTEGER	value
<words lr="" td="" ="" <=""> 4 5 <end_inst td="" <=""> 6 <env td="" <=""> 7 <data td="" <=""> 8 <var_list td="" <=""> 9 10 <var_decl td="" <=""> 11 12 <var_decl_tail td="" <=""> 13 12 <level td="" <=""> 13 12 <level td="" <=""> 14 17 <labels td="" <=""> 16 16 <labels td="" <=""> 19 18 <instruction_list td="" <=""> 21 18 <instruction td="" <=""> 34 34 <expression td="" <=""> 35 35 <expression td="" <=""> 35 35 <expession td="" <=""> 37 38 <exp_and td="" <=""> 38 38 <exp_and td="" <=""> 41 41 <exp_equal td="" <=""> 41 41 <exp_add td="" <=""> 48 48 <exp_add td="" <=""> 48 48 <exp_mult td="" <=""> 52 52 <exp_mult td="" <=""> 58 58 <exp_parenthesis td="" <=""> 59 59 <exp_not td="" <=""> 58 58 <exp_rot td="" <=""></exp_rot></exp_not></exp_parenthesis></exp_mult></exp_mult></exp_add></exp_add></exp_equal></exp_and></exp_and></expession></expression></expression></instruction></instruction_list></labels></labels></level></level></var_decl_tail></var_decl></var_list></data></env></end_inst></words>	<program></program>	1						
WORDS_LR> 4 5 END_INST> 6 ENV> 7 ODATA> 8 <var_list> 9 10 VAR_DECL> 111 11 VAR_DECL_TAIL> 13 12 LEVEL> 11 14 PROC> 14 4 LABELS> 16 14 LABELS_LRS 17 14 INSTRUCTION_LIST> 21 17 INSTRUCTIONS 24 34 ASSIGNATION> 34 34 ASSIGN_END> 34 34 EXPRESSION_LR> 35 35 EXPRESSION_LR> 37 22 EXP_AND 38 38 EXP_EQUAL_N 41 41 EXP_EQUAL_LR> 48 48 EXP_ADD_LR> 48 48 EXP_MULT_LR> 52 52 EXP_PMULT_LR> 59 59 EXP_TERN 61 62 IF 61 62 IF 61 62</var_list>	<IDENT $>$	\parallel 2						
END_INST> 6 <env> 7 <data> 8 <var_list> 9 10 <var_decl> 11 11 <var_decl_tail> 13 12 <level> 14 2PROC> <labels> 16 4 <labels_lr> 17 4 <labels_lr> 19 4 <instruction_list> 21 4 <instruction> 34 34 <assignation> 35 35 <expression_lr> 37 4 <expand_n< td=""> 38 38 <exp_and_lr> 41 41 <exp_equal< td=""> 41 41 <exp_add_lr> 48 48 <exp_add_lr> 52 52 <exp_mult< td=""> 52 52 <exp_mult< td=""> 58 58 <exp_parenthesis> 59 59 <exp_add< td=""> 61 62 <if> 4 6 62 <if> IF END> 6 6 <exp_rot< td=""></exp_rot<></if></if></exp_add<></exp_parenthesis></exp_mult<></exp_mult<></exp_add_lr></exp_add_lr></exp_equal<></exp_and_lr></expand_n<></expression_lr></assignation></instruction></instruction_list></labels_lr></labels_lr></labels></level></var_decl_tail></var_decl></var_list></data></env>	<WORDS $>$		3					
ENV> 7 <data> 8 <var_list> 9 10 VAR_DECL> 11 12 <var_decl_tail> 13 12 <level> 14 44 <proc> 14 44 <labels> 16 44 <labels lr<="" td=""> 17 44 <label> 19 44 <instruction list=""> 21 44 <instruction> 34 34 <assign end=""> 35 35 <expression lr=""> 37 35 <expression lr=""> 37 38 <exp_and lr=""> 41 41 <exp_equal lr=""> 48 48 <exp_add lr=""> 48 48 <exp_add lr=""> 52 52 <exp_mult lr=""> 52 52 <exp_parenthesis> 59 59 <exp_term> 61 62 <if> 41 62 <if> IF END> 62 62 <irstriction 44<="" td="" =""> 44 44</irstriction></if></if></exp_term></exp_parenthesis></exp_mult></exp_add></exp_add></exp_equal></exp_and></expression></expression></assign></instruction></instruction></label></labels></labels></proc></level></var_decl_tail></var_list></data>	<WORDS LR $>$		4	5				
ENV> 7 DATA> 8 VAR_LIST> 9 10 VAR_DECL> 11 12 VAR_DECL_TAIL> 13 12 VAR_DECL_TAIL> 13 12 VAR_DECL_TAIL> 11 14 PROC> 14 14 CLABELS> 16 14 CLABELS LR> 17 17 CLABELS LR 19 11 SINSTRUCTION_LIST> 21 11 SASSIGNATION> 34 34 ASSIGNATION> 35 35 SEXPRESSION_LR> 37 35 EXPRESSION_LR> 37 38 EXP_AND 38 38 EXP_AND 41 41 EXP_EQUAL_LR> 48 48 EXP_ADD 48 48 EXP_MULT_LR> 52 52 EXP_MULT_LR> 59 59 EXP_TERM> 61 62 EIF END> 61 62 EXP_END> 61 62 EXP_EN	<u>—</u>			6				
VAR_LIST> 9 10 <var_decl> 11 11 <var_decl_tail> 13 12 <level> 14 14 <proc> 16 14 <labels_lr> 17 17 <labels_lr> 19 10 <instruction_list> 21 <instruction> 34 34 <assignation> 35 35 <assign_end> 34 34 <expression_lr> 37 38 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_bqual_lr> 48 48 <exp_add_lr> 52 52 <exp_mult> 52 52 <exp_mult_lr> 58 58 <exp_not> 58 58 <exp_pterm> 61 62 <if> 4F 61 62 <if< td=""> END> 61 62 <if< td=""> AND 61 62 <if< td=""> AND 62 62 <if< td=""></if<></if<></if<></if<></if></exp_pterm></exp_not></exp_mult_lr></exp_mult></exp_add_lr></exp_bqual_lr></exp_and_lr></exp_and></expression_lr></assign_end></assignation></instruction></instruction_list></labels_lr></labels_lr></proc></level></var_decl_tail></var_decl>	_				7			
VAR_DECL> 11 <var_decl_tail> 13 12 <uevel> 14 14 PROC> 16 14 <labels> 16 17 <label> 19 17 <label> 19 18 <instruction_list> 21 18 <instruction> 34 34 <assign_end> 35 35 <expression lr=""> 37 35 <expression_lr> 37 38 <exp_and lr=""> 38 38 <exp_and lr=""> 41 41 <exp_equal lr=""> 48 48 <exp_add lr=""> 52 52 <exp_mult lr=""> 52 52 <exp_mult_lr> 58 58 <exp_not> 58 58 <exp_parenthesis> 59 59 <exp_term> 61 62 <if>_END> (CALL_TAIL) (CALL_TAIL) <crad> (WRITE) (CALL_TAIL)</crad></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr></exp_mult></exp_add></exp_equal></exp_and></exp_and></expression_lr></expression></assign_end></instruction></instruction_list></label></label></labels></uevel></var_decl_tail>	<data></data>					8		
VAR_DECL> 11 <var_decl_tail> 13 12 <uevel> 14 14 PROC> 16 14 <labels> 16 17 <label> 19 17 <label> 19 18 <instruction_list> 21 18 <instruction> 34 34 <assign_end> 35 35 <expression lr=""> 37 35 <expression_lr> 37 38 <exp_and lr=""> 38 38 <exp_and lr=""> 41 41 <exp_equal lr=""> 48 48 <exp_add lr=""> 52 52 <exp_mult lr=""> 52 52 <exp_mult_lr> 58 58 <exp_not> 58 58 <exp_parenthesis> 59 59 <exp_term> 61 62 <if>_END> (CALL_TAIL) (CALL_TAIL) <crad> (WRITE) (CALL_TAIL)</crad></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr></exp_mult></exp_add></exp_equal></exp_and></exp_and></expression_lr></expression></assign_end></instruction></instruction_list></label></label></labels></uevel></var_decl_tail>	<var list=""></var>					9		10
VAR_DECL_TAIL> 13 12 <level> 14 PROC> 16 <labels lr=""> 17 <labels 19<="" td="" =""> 19 <instruction_list> 21 <instruction> 34 <assign_end> 35 <xssign_end> 35 <expression lr=""> 37 <exp_and> 38 <exp_and lr=""> 41 <exp_equal< td=""> 41 <exp_equal_lr> 48 <exp_add lr=""> 52 <exp_mult> 52 <exp_mult_lr> 52 <exp_not> 58 <exp_parenthesis> 59 <exp_term> 61 <if _="" end=""> 62 <if _="" end=""> 62 <url _="" tail=""> <read></read></url></if></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr></exp_mult></exp_add></exp_equal_lr></exp_equal<></exp_and></exp_and></expression></xssign_end></assign_end></instruction></instruction_list></labels></labels></level>	_					11		
<level> 14 <proc> 16 <labels_lr> 17 <labels_< td=""> 19 <instruction_list> 21 <instruction> 24 <assignation> 34 34 <assign_end> 35 35 <expression_lr> 37 37 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_equal< td=""> 41 41 <exp_add> 48 48 <exp_add_lr> 52 52 <exp_mult< td=""> 52 52 <exp_mult_lr> 58 58 <exp_parenthesis> 59 59 <exp_term> 61 62 <if>_END> 61 62 <call_tail> <crad></crad></call_tail></if></exp_term></exp_parenthesis></exp_mult_lr></exp_mult<></exp_add_lr></exp_add></exp_equal<></exp_and_lr></exp_and></expression_lr></assign_end></assignation></instruction></instruction_list></labels_<></labels_lr></proc></level>				13				12
<proc> <iabels> 16 <labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assignation> 34 <assign_end> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 38 <exp_equal> 41 <exp_equal_lr> 48 <exp_add> 48 <exp_mult> 52 <exp_mult_lr> 52 <exp_mult_lr> 58 <exp_parenthesis> 59 <exp_term> 61 62 <if> <call< td=""> <read></read></call<></if></exp_term></exp_parenthesis></exp_mult_lr></exp_mult_lr></exp_mult></exp_add></exp_equal_lr></exp_equal></exp_and_lr></exp_and></expression_lr></assign_end></assignation></instruction></instruction_list></label></labels_lr></iabels></proc>							14	
<labels> 16 <labels lr=""> 17 <label> 19 <instruction list=""> 21 <instruction> 34 <assign end=""> 34 <expression lr=""> 35 <expand> 38 <exp_and lr=""> 37 <exp_equal> 41 41 <exp_equal lr=""> 48 48 <exp_add lr=""> 52 52 <exp_mult lr=""> 52 52 <exp_mult_lr> 58 58 <exp_parenthesis> 59 59 <if> 61 62 <if _="" end=""> CALL _ CALL _ <call _="" tail=""> CRAD> CRAD> <write> WRITE> CALC</write></call></if></if></exp_parenthesis></exp_mult_lr></exp_mult></exp_add></exp_equal></exp_equal></exp_and></expand></expression></assign></instruction></instruction></label></labels></labels>								
<labels_lr> 17 <label> 19 <instruction_list> 21 <instruction> 34 <assignation> 34 <assign_end> 35 <assign_end> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal_lr> 48 <exp_add_lr> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 58 <exp_parenthesis> 59 <exp_term> 61 <if>_END> 61 <call_< td=""> 41 <exall> 62 <if< td=""> 61 <call_tail> 62 <write> 61</write></call_tail></if<></exall></call_<></if></exp_term></exp_parenthesis></exp_mult_lr></exp_mult></exp_add_lr></exp_add_lr></exp_equal_lr></exp_and_lr></exp_and></expression_lr></assign_end></assign_end></assignation></instruction></instruction_list></label></labels_lr>			16					
<label> 19 <instruction_list> 21 <instruction> 34 <assign_end> 34 <expression> 35 <expression_lr> 37 <exp_and> 38 <exp_and_lr> 41 <exp_equal_lr> 41 <exp_add> 48 <exp_add_lr> 52 <exp_mult> 52 <exp_mult_lr> 58 <exp_parenthesis> 59 <if> 61 <call< td=""> 62 <if_end> 61 <call_tail> CALL_TAIL> <read> WRITE></read></call_tail></if_end></call<></if></exp_parenthesis></exp_mult_lr></exp_mult></exp_add_lr></exp_add></exp_equal_lr></exp_and_lr></exp_and></expression_lr></expression></assign_end></instruction></instruction_list></label>			17					
<instruction list="" td="" ="" <=""> 21 <instruction td="" <=""> 34 34 <assign_end 34<="" td="" =""> 34 34 <expression lr="" td="" ="" <=""> 35 35 <expession_lr 37<="" td="" =""> 38 38 <exp_and lr="" td="" ="" <=""> 41 41 <exp_equal 41<="" td="" =""> 41 41 <exp_add lr="" td="" ="" <=""> 48 48 <exp_add lr="" td="" ="" <=""> 52 52 <exp_mult lr="" td="" ="" <=""> 52 52 <exp_not 58<="" td="" =""> 58 58 <exp_parenthesis 59<="" td="" =""> 59 59 <if end="" td="" ="" <=""> 61 62 <if end="" td="" ="" <=""> 61 62 <if end="" td="" ="" <=""> 61 62 <irad td="" <=""> <write td="" <=""></write></irad></if></if></if></exp_parenthesis></exp_not></exp_mult></exp_add></exp_add></exp_equal></exp_and></expession_lr></expression></assign_end></instruction></instruction>	<u>—</u>		19					
<instruction> <assignation> <assign_end> 34 34 <expression> 35 35 <expression_lr> 37 38 <exp_and> 38 38 <exp_and_lr> 41 41 <exp_equal_lr> 48 48 <exp_add> 48 48 <exp_mult> 52 52 <exp_mult_lr> 52 52 <exp_not> 58 58 <exp_parenthesis> 59 59 <if> 61 62 <if_end> 61 62 <call_tail> <read></read></call_tail></if_end></if></exp_parenthesis></exp_not></exp_mult_lr></exp_mult></exp_add></exp_equal_lr></exp_and_lr></exp_and></expression_lr></expression></assign_end></assignation></instruction>			21					
<assignation> <assign_end> 34 34 <expression> 35 35 <expession_lr> 37 38 38 <exp_and_lr> 41 41 41 <exp_equal_lr> 48 48 48 <exp_add_lr> 52 52 <exp_mult> 52 52 <exp_not> 58 58 <exp_parenthesis> 59 59 <if> 61 62 <if_end> 61 62 <call> CALL_TAIL> <read></read></call></if_end></if></exp_parenthesis></exp_not></exp_mult></exp_add_lr></exp_equal_lr></exp_and_lr></expession_lr></expression></assign_end></assignation>	_							
<assign_end> 34 34 <expression> 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_lr> 52 52 <exp_not> 58 58 <exp_parenthesis> 59 59 <if> 61 62 <if_end> 61 62 <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if></exp_parenthesis></exp_not></exp_mult_lr></exp_add_lr></exp_equal_lr></exp_and_lr></exp_and></expression_lr></expression></assign_end>								
EXPRESSION 35 35 EXPRESSION LR > 37 EXP AND LR > 38 EXP EQUAL 41 41 EXP EQUAL LR > 48 48 EXP ADD LR > 52 52 EXP MULT LR > 58 58 EXP NOT > 58 58 EXP PARENTHESIS > 59 59 EXP TERM > 61 62 IF END > CALL CALL TAIL > CALL TAIL > KEAD > WRITE > CALL CALL			34				34	
EXPRESSION_LR> 37 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> 58 58 EXP_NOT> 58 58 EXP_PARENTHESIS> 59 59 EXP_TERM> 61 62 IF_END> 62 61 CALL> CALL_TAIL> CALL_TAIL> KEAD> WRITE> CALL_TAIL>	_		35				35	
EXP_AND> 38 EXP_AND_LR> 41 EXP_EQUAL> 41 EXP_EQUAL_LR> 48 EXP_ADD> 48 EXP_MULT> 52 EXP_MULT_LR> 58 EXP_NOT> 58 EXP_PARENTHESIS> 59 EXP_TERM> 61 IF_END> 62 CALL> CALL_TAIL> READ> WRITE>				37				
EXP_AND_LR> EXP_EQUAL> 41 41 EXP_EQUAL_LR> 48 48 EXP_ADD> 48 48 EXP_ADD_LR> 52 52 EXP_MULT_LR> 58 58 EXP_NOT> 58 59 EXP_TERM> 61 62 IF_END> 62 CALL> CALL_TAIL> <read> WRITE></read>	-		38				38	
EXP_EQUAL> 41 41 EXP_EQUAL_LR> 48 48 EXP_ADD_LR> 52 52 EXP_MULT> 52 52 EXP_MULT_LR> 58 58 EXP_NOT> 58 59 EXP_PARENTHESIS> 59 59 EXP_TERM> 61 62 IF_END> 62 62 CALL> CALL_TAIL> CALL_TAIL> <read> WRITE> CALL_TAIL</read>								
EXP_EQUAL_LR> EXP_ADD> 48 EXP_ADD_LR> 52 EXP_MULT> 52 EXP_NOT> 58 EXP_PARENTHESIS> 59 EXP_TERM> 61 IF_END> 62 CALL CALL_TAIL> READ> WRITE>			41				41	
EXP_ADD> 48 48 EXP_ADD_LR> 52 52 EXP_MULT> 52 52 EXP_MULT_LR> 58 58 EXP_NOT> 58 59 EXP_PARENTHESIS> 59 59 EXP_TERM> 61 62 IF> 61 62 CALL> CALL_TAIL> CALL_TAIL> READ> CWRITE> CALL_TAIL>								
<exp_add_lr> <exp_mult> 52 <exp_mult_lr> 58 <exp_not> 58 <exp_parenthesis> 59 <exp_term> 61 62 <if> <if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr></exp_mult></exp_add_lr>			48				48	
<exp_mult> 52 <exp_mult_lr> 58 <exp_not> 58 <exp_parenthesis> 59 <exp_term> 61 <if> 62 <if_end> 62 <call> <read> <write> 52 52 52 52 52 58 58 59 59 61 62</write></read></call></if_end></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr></exp_mult>								
<exp_mult_lr> <exp_not> 58 <exp_parenthesis> 59 <exp_term> 61 <if> <if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if></exp_term></exp_parenthesis></exp_not></exp_mult_lr>			52				52	
<exp_not> 58 58 <exp_parenthesis> 59 59 <exp_term> 61 62 <if> <if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if></exp_term></exp_parenthesis></exp_not>	<u>—</u>							
<exp_parenthesis> 59 <exp_term> 61 <if> <if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if></exp_term></exp_parenthesis>			58				58	
<exp_term> 61 <if> 62 <if_end> 62 <call> <call_tail> <read> <write> 62</write></read></call_tail></call></if_end></if></exp_term>								
<if> <if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end></if>	_							
<if_end> <call> <call_tail> <read> <write></write></read></call_tail></call></if_end>	$<$ IF $>$ $^-$							
<call> <call_tail> <read> <write></write></read></call_tail></call>								
<call_tail> <read> <write></write></read></call_tail>								
<read> <write></write></read>								
<write></write>	_							
_			73				73	
	_	П						

	procedure	move	compute	add	substract	multiply	divide
<program></program>			-				
<IDENT $>$							
<words></words>							
<words lr=""></words>							
<end inst=""></end>							
<env></env>							
<data></data>							
<var list=""></var>	10						
<var decl=""></var>	10						
<var decl="" tail=""></var>							
<level></level>							
<proc></proc>	15						
<labels></labels>	10						
<labels lr=""></labels>							
<label></label>							
<instruction list=""></instruction>		20	20	20	20	20	20
<instruction_list></instruction_list>		22	22	22	$\frac{20}{22}$	22	$\frac{20}{22}$
<assignation></assignation>		28	29	30	31	$\frac{22}{32}$	33
<assign end=""></assign>		20	23	30	31	52	55
<expression></expression>							
<expression lr=""></expression>							
<exp and=""></exp>							
<exp and="" lr=""></exp>							
<exp equal=""></exp>							
<exp equal="" lr=""></exp>							
<exp add=""></exp>							
<exp add="" lr=""></exp>							
<exp mult=""></exp>							
<exp lr="" mult=""></exp>							
<exp_not></exp_not>							
<exp parenthesis=""></exp>							
<exp term=""></exp>							
<if></if>							
<if_end></if_end>							
<call></call>							
<call tail=""></call>							
<read></read>							
<write></write>							
<write_tail></write_tail>							
\WIGHT_IMIL/	I						

	perform	accept	display	stop	_	not	(true	false
<program></program>	-	-		-			`		
<ident></ident>									
<words></words>									
<words lr=""></words>									
<end inst=""></end>									
<env></env>									
<data></data>									
<var list=""></var>									
_									
<var_decl tails<="" td=""><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></var_decl>									
<var_decl_tail></var_decl_tail>									
<level></level>									
<proc></proc>									
<labels></labels>									
<labels_lr></labels_lr>									
<LABEL $>$									
<instruction_list></instruction_list>	20	20	20	20					
<instruction></instruction>	24	25	26	27					
<assignation></assignation>									
$<$ ASSIGN_END $>$					34	34	34	34	34
<expression></expression>					35	35	35	35	35
<expression_lr></expression_lr>									
<exp_and></exp_and>					38	38	38	38	38
<exp_and_lr></exp_and_lr>									
$<$ EXP $_$ EQUAL $>$					41	41	41	41	41
$<$ EXP_EQUAL_LR $>$									
$\langle \text{EXP_ADD} \rangle$					48	48	48	48	48
$<$ EXP_ADD_LR $>$					50				
$<$ EXP $_$ MULT $>$					52	52	52	52	52
<exp lr="" mult=""></exp>					55				
<exp not=""></exp>					56	57	58	58	58
<exp parenthesis=""></exp>							59	60	60
<exp_term></exp_term>								63	64
<IF $>$									
<if_end></if_end>									
$<$ C $\overline{\text{CALL}}>$	68								
<call tail=""></call>									
<read></read>		71							
<write></write>		. –	72						
<write_tail></write_tail>			. –		73	73	73	73	73
	I				. ~	. •	. •	. •	. •

	or	and	=	<	>	<=	>=	*	/	if	else	end-if	+	until
<PROGRAM $>$,					
<IDENT $>$														
<WORDS $>$														
<words lr=""></words>														
<end inst=""></end>														
<env></env>														
<data></data>														
<var list=""></var>														
<var_decl></var_decl>														
<var_decl tail=""></var_decl>														
<level></level>														
<proc></proc>														
<labels></labels>														
<labels lr=""></labels>														
<label></label>														
<instruction list=""></instruction>										20				
<instruction_list></instruction_list>										23				
<assignation></assignation>										23				
<assignation></assignation>														

```
until STRING , to from giving ) then
<PROGRAM>
<IDENT>
<WORDS>
<WORDS LR>
<END_INST>
<ENV>
<DATA>
<VAR LIST>
<VAR_DECL>
<VAR DECL TAIL>
<LEVEL>
<PROC>
<LABELS>
<LABELS LR>
<LABEL>
<INSTRUCTION_LIST>
<INSTRUCTION>
<ASSIGNATION>
<ASSIGN END>
<EXPRESSION>
<EXPRESSION_LR>
                                   37 37
                                           37
                                                37
                                                     37
                                                         37
\langle \text{EXP\_AND} \rangle
<EXP AND LR>
<EXP\_EQU\overline{A}L>
<EXP_EQUAL_LR>
\langle \text{EXP\_ADD} \rangle
<EXP_ADD_LR>
<EXP\_MULT>
<EXP NOT>
<EXP PARENTHESIS>
<EXP_TERM>
<IF>
<IF_END>
<CALL>
<CALL_TAIL>
                      69
<READ>
<WRITE>
<WRITE_TAIL>
                             74
```