```
[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	$\mathrm{First}^1$	$Follow^1$
<program></program>	identification	
<IDENT $>$	identification	environment
<WORDS $>$	ID	
$<$ WORDS_LR $>$	ID, $\varepsilon$	

<end_inst></end_inst>		program-id, date-written, environment, configuration, source-computer, object-computer, data, working-storage, INTEGER, $\varepsilon$ , ID, move, compute, add, substract, multiply, divide, if, perform, accept, display, stop
<env></env>	environment	data
<data></data>	data	procedure
<var list=""></var>	INTEGER, $\varepsilon$	<del>-</del>
<var_list></var_list>	INTEGER, E INTEGER	procedure
<del>_</del>		INTEGER, $\varepsilon$
<var_decl_tail></var_decl_tail>	value, .	INTEGER, $\varepsilon$
<level></level>	INTEGER	ID
<proc></proc>	procedure	1
<labels></labels>	ID	end
<labels_lr></labels_lr>	$ID, \varepsilon$	end
<label></label>	ID	·
<instruction_list></instruction_list>	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, $\varepsilon$	ID, $\varepsilon$
<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, $\varepsilon$
<assignation></assignation>	move, compute, add, substract,	move, compute, add, substract, mul-
(11001011111011)	multiply, divide	tiply, divide, if, perform, accept, dis-
	maroipi, arvide	play, stop, $\varepsilon$
<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, $\varepsilon$	to, ., from, , , giving, ), then
<exp and=""></exp>	-, not, (, ID, INTEGER, true, false	or, $\varepsilon$
<exp and="" lr=""></exp>	and, $\varepsilon$	or, $\varepsilon$
<exp equal=""></exp>	-, not, (, ID, INTEGER, true, false	and, $\varepsilon$
<u> </u>		_^
<exp_equal_lr> <exp_add></exp_add></exp_equal_lr>	$=,<,>,<=,>=,\varepsilon$ -, not, (, ID, INTEGER, true, false	and, $\varepsilon$
<exp_add_lr></exp_add_lr>	$+$ , $-$ , $\varepsilon$	
<exp mult=""></exp>		$=,<,>,<=,>=,\varepsilon$
<u> </u>	-, not, (, ID, INTEGER, true, false	$+$ , -, $\varepsilon$
<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, $\varepsilon$
<if_end></if_end>	else, end-if	move, compute, add, substract, mul-
		tiply, divide, if, perform, accept, dis-
		play, stop, $\varepsilon$

<CALL>perform move, compute, add, substract, multiply, divide, if, perform, accept, display, stop,  $\varepsilon$ <CALL TAIL>until, . move, compute, add, substract, multiply, divide, if, perform, accept, display, stop,  $\varepsilon$ <READ> move, compute, add, substract, mulaccept tiply, divide, if, perform, accept, display, stop,  $\varepsilon$ <WRITE>display move, compute, add, substract, multiply, divide, if, perform, accept, display, stop,  $\varepsilon$ <WRITE\_TAIL>STRING, -, not, (, ID, INTEGER, move, compute, add, substract, multrue, false tiply, divide, if, perform, accept, display, stop,  $\varepsilon$ 

PROGRAM>       1         IDENT>       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       14         PROC>       16       17         LABELS_LR>       17       18         LABEL>       19       10         INSTRUCTION_LIST>       21       11         ASSIGNATION>       34       34         ASSIGN_END>       35       35         EXPRESSION_LR>       37       38         EXP_AND_LR>       38       38		identification	ID		environment	data	INTEGER	value
<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_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assign_end>       34         <expression_lr>       35         <expression_lr>       37         <exp_and< td="">       38         <exp_and_lr></exp_and_lr></exp_and<></expression_lr></expression_lr></assign_end></instruction></instruction_list></label></labels_lr></proc></level></var_decl_tail></var_decl></var_list></data></env></end_inst></words_lr></words>	<program></program>	1						
<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       14         <proc>       16       17       18         <labels_lr>       17       19       10         <instruction_list>       21       11       11         <instruction>       21       34       34       34         <expression>       35       35       35         <expression_lr>       37       38       38         <exp_and_lr>       38       38</exp_and_lr></expression_lr></expression></instruction></instruction_list></labels_lr></proc></level></var_decl_tail></var_decl></var_list></data></env></end_inst></words_lr>	<IDENT $>$	$\parallel$ 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         <assign_end>       35       35         <expression_lr>       37       37         <exp_and< td="">       38       38         <exp_and_lr>       38       38</exp_and_lr></exp_and<></expression_lr></assign_end></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         <var_decl_tail>       13       12         <level>       14         <proc>       16         <labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       21         <assignation>       34       34         <expression 35<="" td=""  ="">       35         <expression_lr>       37         <exp_and 38<="" td=""  ="">       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label></labels_lr></proc></level></var_decl_tail></var_decl></var_list></data></env>	<WORDS LR $>$		4	5				
<env>       7         <data>       8         <var_list>       9       10         <var_decl>       11       12         <var_decl_tail>       13       12         <level>       14       14         <proc>       16       17         <labels_lr>       17       17         <label>       19       11         <instruction_list>       21       11         <instruction>       21       34         <assign_end>       34       34         <expression_lr>       37       35         <exp_and>       38       38         <exp_and_lr>       38       38</exp_and_lr></exp_and></expression_lr></assign_end></instruction></instruction_list></label></labels_lr></proc></level></var_decl_tail></var_decl></var_list></data></env>	<u>—</u>			6				
<var_list>       9       10         <var_decl>       11       12         <var_decl_tail>       13       12         <level>       14       14         <proc>       16       17         <labels_lr>       17       17         <label>       19       10         <instruction>       21       31         <assignation>       34       34         <assign_end>       35       35         <expression_lr>       37       37         <exp_and>       38       38         <exp_and_lr>       38       38</exp_and_lr></exp_and></expression_lr></assign_end></assignation></instruction></label></labels_lr></proc></level></var_decl_tail></var_decl></var_list>	<del>_</del>				7			
<var_decl>       11         <var_decl_tail>       13       12         <level>       14       14         <proc>       16       17       17         <labels_lr>       17       19       11         <instruction_list>       21       19         <instruction>       34       34         <assignation>       35       35         <expression_lr>       37       35         <exp_and>       38       38         <exp_and_lr>       38       38</exp_and_lr></exp_and></expression_lr></assignation></instruction></instruction_list></labels_lr></proc></level></var_decl_tail></var_decl>	<data></data>					8		
<var_decl>       11         <var_decl_tail>       13       12         <level>       14       14         <proc>       16       17       17         <labels_lr>       17       19       11         <instruction_list>       21       19         <instruction>       34       34         <assignation>       35       35         <expression_lr>       37       35         <exp_and>       38       38         <exp_and_lr>       38       38</exp_and_lr></exp_and></expression_lr></assignation></instruction></instruction_list></labels_lr></proc></level></var_decl_tail></var_decl>	<var list=""></var>					9		10
<var_decl_tail>       13       12         <level>       14         <proc>       16         <labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression s<="" td="">       35         <expression_lr>       37         <exp_and s<="" td="">       38         <exp_and lr="">       38</exp_and></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label></labels_lr></proc></level></var_decl_tail>	<del>_</del>					11		
<level>       14         <proc>       16         <labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label></labels_lr></proc></level>	<del></del>			13				12
<proc>       16         <labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label></labels_lr></proc>							14	
<labels>       16         <labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label></labels_lr></labels>								
<labels_lr>       17         <label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression_end>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression_end></assignation></instruction></instruction_list></label></labels_lr>			16					
<label>       19         <instruction_list>       21         <instruction>       34         <assignation>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list></label>			17					
<instruction_list>       21         <instruction>       34         <assignation>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assignation></instruction></instruction_list>	<u>—</u>		19					
<instruction> <assignation> <assign_end>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assign_end></assignation></instruction>			21					
<assignation> <assign_end>       34       34         <expression>       35       35         <expression_lr>       37       38       38         <exp_and_lr>       38       38</exp_and_lr></expression_lr></expression></assign_end></assignation>	<del>_</del>							
<assign_end>       34         <expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression></assign_end>								
<expression>       35         <expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr></expression>			34				34	
<expression_lr>       37         <exp_and>       38         <exp_and_lr>       38</exp_and_lr></exp_and></expression_lr>	<del>_</del>		35				35	
<exp_and> 38 38 38   <exp_and_lr></exp_and_lr></exp_and>				37				
<exp_and_lr></exp_and_lr>	<del>-</del>		38				38	
$\langle \text{EXP EQUAL} \rangle$   41	$<$ EXP $^-$ EQU $^-$ L $>$		41				41	
<exp equal="" lr=""></exp>								
$\langle \text{EXP\_ADD} \rangle$ 48			48				48	
<exp add="" lr=""></exp>	<del></del>							
$\langle \text{EXP}^{-}\text{MULT} \rangle$ 52			52				52	
<exp lr="" mult=""></exp>	<u>—</u>							
<exp_not> 56 56</exp_not>			56				56	
<exp parenthesis="">   59</exp>	<del></del>							
<exp_term> 61</exp_term>	_							
${<}\mathrm{IF}{>}$	$<$ IF $>$ $^-$							
<if_end></if_end>								
<call></call>								
<call tail=""></call>								
<read></read>	_							
<write></write>								
<write_tail> 73</write_tail>			73				73	
— n	_	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	$\frac{20}{22}$	$\frac{20}{22}$	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 $\_$ 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					
<assign_end></assign_end>														
<expression></expression>	36													
<expression_lr></expression_lr>	30													
<exp_and></exp_and>	40	20												
<exp_and_lr></exp_and_lr>	40	39												
<exp_equal></exp_equal>		47	40	49	4.4	45	16							
<exp_equal_lr></exp_equal_lr>		47	42	43	44	45	46							
<exp_add></exp_add>			F 1	F-1	F 1	F 1	F-1						40	
<exp_add_lr></exp_add_lr>			51	51	51	51	51						49	
<exp_mult></exp_mult>									<b>-</b> ,					
<exp_mult_lr></exp_mult_lr>								53	54				55	
<exp_not></exp_not>														
<exp_parenthesis></exp_parenthesis>														
$\langle \text{EXP\_TERM} \rangle$														
<if></if>										65				
$\langle IF\_END \rangle$											66	67		
<call></call>														
<call_tail></call_tail>														69
<read></read>														
<write></write>														
$<$ WRITE_TAIL $>$														

```
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
```