[1]	<PROGRAM>	→	<IDENT><ENV><DATA><PROC>
[2]	<IDENT>	→	identification division <END_INST> program-id. ID<END_INST> author. WORDS<END_INST> date-written. WORDS <END_INST>
[3]	<WORDS>	→	ID <WORDS_LR>
[4]	<WORDS_LR>	→	ID <WORDS_LR>
[5]		→	ε
[6]	<END_INST>	→	.\n
[7]	<ENV>	→	environment division<END_INST> configuration 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]		→	ε
[11]	<VAR_DECL>	→	<LEVEL> ID pic IMAGE <VAR_DECL_TAIL>
[12]	<VAR_DECL_TAIL>	→	value INTEGER<END_INST>
[13]		→	<END_INST>
[14]	<LEVEL>	→	INTEGER
[15]	<PROC>	→	procedure division<END_INST> ID section<END_INST> <LABELS> end program ID.
[16]	<LABELS>	→	<LABEL><END_INST> <INSTRUCTION_LIST> <LABELS_LR>
[17]	<LABELS_LR>	→	<LABEL><END_INST> <INSTRUCTION_LIST> <LABELS_LR>
[18]		→	ε
[19]	<LABEL>	→	ID
[20]	<INSTRUCTION_LIST>	→	<INSTRUCTION> <INSTRUCTION_LIST>
[21]		→	ε
[22]	<INSTRUCTION>	→	<ASSIGNATION>
[23]		→	<IF>
[24]		→	<CALL>
[25]		→	<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>
[31]		→	subtract <EXPRESSION> from ID<END_INST>
[32]		→	multiply <ASSIGN_END><END_INST>
[33]		→	divide <ASSIGN_END><END_INST>
[34]	<ASSIGN_END>	→	<EXPRESSION>,<EXPRESSION> giving ID
[35]	<EXPRESSION>	→	<EXP_AND> <EXPRESSION_LR>
[36]	<EXPRESSION_LR>	→	or <EXP_AND> <EXPRESSION_LR>
[37]		→	ε
[38]	<EXP_AND>	→	<EXP_EQUAL> <EXP_AND_LR>
[39]	<EXP_AND_LR>	→	and <EXP_EQUAL> <EXP_AND_LR>

[40]		→	ε
[41]	<EXP_EQUAL>	→	<EXP_ADD> <EXP_EQUAL_LR>
[42]	<EXP_EQUAL_LR>	→	= <EXP_ADD> <EXP_EQUAL_LR>
[43]		→	< <EXP_ADD> <EXP_EQUAL_LR>
[44]		→	> <EXP_ADD> <EXP_EQUAL_LR>
[45]		→	<= <EXP_ADD> <EXP_EQUAL_LR>
[46]		→	>= <EXP_ADD> <EXP_EQUAL_LR>
[47]		→	ε
[48]	<EXP_ADD>	→	<EXP_MULT> <EXP_ADD_LR>
[49]	<EXP_ADD_LR>	→	+ <EXP_MULT> <EXP_ADD_LR>
[50]		→	- <EXP_MULT> <EXP_ADD_LR>
[51]		→	ε
[52]	<EXP_MULT>	→	<EXP_NOT> <EXP_MULT_LR>
[53]	<EXP_MULT_LR>	→	* <EXP_NOT> <EXP_MULT_LR>
[54]		→	/ <EXP_NOT> <EXP_MULT_LR>
[55]		→	ε
[56]	<EXP_NOT>	→	-<EXP_NOT>
[57]		→	not <EXP_NOT>
[58]		→	<EXP_PARENTHESIS>
[59]	<EXP_PARENTHESIS>	→	(<EXPRESSION>)
[60]		→	<EXP_TERM>
[61]	<EXP_TERM>	→	ID
[62]		→	INTEGER
[63]		→	true
[64]		→	false
[65]	<IF>	→	if <EXPRESSION> then <INSTRUCTION_LIST> <IF_END>
[66]	<IF_END>	→	else <INSTRUCTION_LIST> end-if
[67]		→	end-if
[68]	<CALL>	→	perform ID <CALL_TAIL>
[69]	<CALL_TAIL>	→	until <EXPRESSION><END_INST>
[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>	identification	
<IDENT>	identification	environment
<WORDS>	ID	.
<WORDS_LR>	ID, ε	.

<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>	environment	data
<DATA>	data	procedure
<VAR_LIST>	INTEGER, ε	procedure
<VAR_DECL>	INTEGER	INTEGER, ε
<VAR_DECL_TAIL>	value, .	INTEGER, ε
<LEVEL>	INTEGER	ID
<PROC>	procedure	
<LABELS>	ID	end
<LABELS_LR>	ID, ε	end
<LABEL>	ID	.
<INSTRUCTION_LIST>	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε	ID, ε
<INSTRUCTION>	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε
<ASSIGNATION>	move, compute, add, substract, multiply, divide	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε
<ASSIGN_END>	-, not, (, ID, INTEGER, true, false	.
<EXPRESSION>	-, not, (, ID, INTEGER, true, false	to, ., from, , , giving,), then
<EXPRESSION_LR>	or, ε	to, ., from, , , giving,), then
<EXP_AND>	-, not, (, ID, INTEGER, true, false	or, ε
<EXP_AND_LR>	and, ε	or, ε
<EXP_EQUAL>	-, not, (, ID, INTEGER, true, false	and, ε
<EXP_EQUAL_LR>	=, <, >, <=, >=, ε	and, ε
<EXP_ADD>	-, not, (, ID, INTEGER, true, false	=, <, >, <=, >=, ε
<EXP_ADD_LR>	+, -, ε	=, <, >, <=, >=, ε
<EXP_MULT>	-, not, (, ID, INTEGER, true, false	+, -, ε
<EXP_MULT_LR>	*, /, ε	+, -, ε
<EXP_NOT>	-, not, (, ID, INTEGER, true, false	*, /, ε
<EXP_PARENTHESIS>	(, ID, INTEGER, true, false	*, /, ε
<EXP_TERM>	ID, INTEGER, true, false	*, /, ε
<IF>	if	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε
<IF_END>	else, end-if	move, compute, add, substract, multiply, divide, if, perform, accept, display, 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>	accept	move, compute, add, substract, multiply, 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, true, false	move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε

	identification	ID	.	environment	data	INTEGER	value
<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	
<PROC>							
<LABELS>		16					
<LABELS_LR>		17					
<LABEL>		19					
<INSTRUCTION_LIST>		21					
<INSTRUCTION>							
<ASSIGNATION>							
<ASSIGN_END>		34				34	
<EXPRESSION>		35				35	
<EXPRESSION_LR>			37				
<EXP_AND>		38				38	
<EXP_AND_LR>							
<EXP_EQUAL>		41				41	
<EXP_EQUAL_LR>							
<EXP_ADD>		48				48	
<EXP_ADD_LR>							
<EXP_MULT>		52				52	
<EXP_MULT_LR>							
<EXP_NOT>		56				56	
<EXP_PARENTHESIS>		59				59	
<EXP_TERM>		61				62	
<IF>							
<IF_END>							
<CALL>							
<CALL_TAIL>							
<READ>							
<WRITE>							
<WRITE_TAIL>		73				73	

	procedure	move	compute	add	substract	multiply	divide
<PROGRAM>							
<IDENT>							
<WORDS>							
<WORDS_LR>							
<END_INST>							
<ENV>							
<DATA>							
<VAR_LIST>	10						
<VAR_DECL>							
<VAR_DECL_TAIL>							
<LEVEL>							
<PROC>	15						
<LABELS>							
<LABELS_LR>							
<LABEL>							
<INSTRUCTION_LIST>		20	20	20	20	20	20
<INSTRUCTION>		22	22	22	22	22	22
<ASSIGNATION>		28	29	30	31	32	33
<ASSIGN_END>							
<EXPRESSION>							
<EXPRESSION_LR>							
<EXP_AND>							
<EXP_AND_LR>							
<EXP_EQUAL>							
<EXP_EQUAL_LR>							
<EXP_ADD>							
<EXP_ADD_LR>							
<EXP_MULT>							
<EXP_MULT_LR>							
<EXP_NOT>							
<EXP_PARENTHESIS>							
<EXP_TERM>							
<IF>							
<IF_END>							
<CALL>							
<CALL_TAIL>							
<READ>							
<WRITE>							
<WRITE_TAIL>							

	perform	accept	display	stop	-	not	(	true	false
<PROGRAM>									
<IDENT>									
<WORDS>									
<WORDS_LR>									
<END_INST>									
<ENV>									
<DATA>									
<VAR_LIST>									
<VAR_DECL>									
<VAR_DECL_TAIL>									
<LEVEL>									
<PROC>									
<LABELS>									
<LABELS_LR>									
<LABEL>									
<INSTRUCTION_LIST>	20	20	20	20					
<INSTRUCTION>	24	25	26	27					
<ASSIGNATION>									
<ASSIGN_END>					34	34	34	34	34
<EXPRESSION>					35	35	35	35	35
<EXPRESSION_LR>									
<EXP_AND>					38	38	38	38	38
<EXP_AND_LR>									
<EXP_EQUAL>					41	41	41	41	41
<EXP_EQUAL_LR>									
<EXP_ADD>					48	48	48	48	48
<EXP_ADD_LR>					50				
<EXP_MULT>					52	52	52	52	52
<EXP_MULT_LR>					55				
<EXP_NOT>					56	57	58	58	58
<EXP_PARENTHESIS>							59	60	60
<EXP_TERM>								63	64
<IF>									
<IF_END>									
<CALL>	68								
<CALL_TAIL>									
<READ>		71							
<WRITE>			72						
<WRITE_TAIL>					73	73	73	73	73

	or	and	=	<	>	<=	>=	*	/	if	else	end-if	+	until
<PROGRAM>														
<IDENT>														
<WORDS>														
<WORDS_LR>														
<END_INST>														
<ENV>														
<DATA>														
<VAR_LIST>														
<VAR_DECL>														
<VAR_DECL_TAIL>														
<LEVEL>														
<PROC>														
<LABELS>														
<LABELS_LR>														
<LABEL>														
<INSTRUCTION_LIST>									20					
<INSTRUCTION>									23					
<ASSIGNATION>														
<ASSIGN_END>														
<EXPRESSION>														
<EXPRESSION_LR>	36													
<EXP_AND>														
<EXP_AND_LR>	40	39												
<EXP_EQUAL>														
<EXP_EQUAL_LR>		47	42	43	44	45	46							
<EXP_ADD>														
<EXP_ADD_LR>			51	51	51	51	51						49	
<EXP_MULT>														
<EXP_MULT_LR>								53	54				55	
<EXP_NOT>														
<EXP_PARENTHESIS>														
<EXP_TERM>														
<IF>										65				
<IF_END>											66	67		
<CALL>														
<CALL_TAIL>														69
<READ>														
<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
<EXP_AND>								
<EXP_AND_LR>								
<EXP_EQUAL>								
<EXP_EQUAL_LR>								
<EXP_ADD>								
<EXP_ADD_LR>								
<EXP_MULT>								
<EXP_MULT_LR>								
<EXP_NOT>								
<EXP_PARENTHESIS>								
<EXP_TERM>								
<IF>								
<IF_END>								
<CALL>								
<CALL_TAIL>	69							
<READ>								
<WRITE>								
<WRITE_TAIL>		74						