```
[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> <INSTRUCTIONLIST>
[21]
[22]
     <INSTRUCTION>
                                <ASSIGNATION>
[23]
                             \rightarrow <IF>
[24]
                             \rightarrow <CALL>
                             \rightarrow <READ>
[25]
[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 <ASSING END><END INST>
[33]
                                 divide <ASSING_END><END_INST>
[34]
    <ASSIGN END>
                                 <EXPRESSION>,<EXPRESSION> giving ID

ightarrow <EXP AND> <EXPRESSION LR>
[35]
     <EXPRESSION>
[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]
                             \rightarrow \varepsilon
[41]
     <EXP EQUAL>
                             \rightarrow <EXP ADD> <EXP EQUAL LR>
     <EXP_EQUAL LR>
[42]
                             \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>
                                >= \langle \text{EXP ADD} \rangle \langle \text{EXP EQUAL LR} \rangle
[46]
[47]
     <EXP ADD>
                                <EXP MULT> <EXP ADD LR>
[48]
     <EXP\_ADD\_LR>
                                + <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>
                                -<EXP NOT>
[57]
                                not <EXP_NOT>
[58]
                                <EXP PARENTHESIS>
[59]
     <EXP PARENTHESIS>
                                (<EXPRESSION>)
[60]
                                <EXP TERM>
     <\!\!\mathrm{EXP\_TERM}\!\!>
[61]
                                ID
[62]
                                INTEGER
                             \rightarrow
[63]
                                true
[64]
                                false
    \langle IF \rangle
                                if <EXPRESSION> then <INSTRUCTION_LIST>
[65]
                                 <IF END>
     <IF END>
[66]
                                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>
                                display <WRITE TAIL>
[72]
     <WRITE>
[73]
     <WRITE_TAIL>
                                <EXPRESSION><END_INST>
[74]
                             → STRING<END INST>
```

Variable	First^1	Follow^1	
<program></program>	identification		
<IDENT $>$	identification		
<WORDS $>$	ID		
$<$ WORDS_LR $>$	ID, ε		
$<\!\!\mathrm{END_INST}\!\!>$			
<ENV $>$	${ m environment}$		
<DATA $>$	data		
$<$ VAR_LIST $>$	INTEGER, ε		
$<$ VAR_DECL $>$	INTEGER		
<var decl="" tail=""></var>	value, .		

<LEVEL>**INTEGER** <PROC> procedure <LABELS> ID<LABELS LR> ID, ε <LABEL> ID <INSTRUCTION LIST> move, compute, add, substract, multiply, divide, if, perform, accept, display, stop, ε <INSTRUCTION> move, compute, add, substract, multiply, divide, if, perform, accept, display, stop <ASSIGNATION> move, compute, add, substract, multiply, divide <ASSIGN END> -, not, (, ID, INTEGER, true, false <EXPRESSION> -, not, (, ID, INTEGER, true, false <EXPRESSION LR> <EXP AND>-, not, (, ID, INTEGER, true, false <EXP_AND_LR> <EXP EQUAL> -, not, (, ID, INTEGER, true, false <EXP_EQUAL_LR> =, <, >, <=, >=, ε <EXP ADD>-, not, (, ID, INTEGER, true, false <EXP ADD LR> +, -, ε <EXP $_$ MULT>-, not, (, ID, INTEGER, true, false <EXP MULT LR> *, /, ε <EXP $_$ NOT>-, not, (, ID, INTEGER, true, false (, ID, INTEGER, true, false <EXP PARENTHESIS> <EXP $_$ TERM>ID, INTEGER, true, false <IF>if <IF END> else, end-if perform <CALL> <CALL TAIL> until, . <READ>accept <WRITE> display <WRITE TAIL> STRING, -, not, (, ID, INTEGER, true, false