

# SCOPE MANAGEMENT REPORT

Full Name: Dinh Viet Quang

Student ID: 20215235

1. Results (a part of the screen) received when executing project with example3.kpl.

```
TERMINAL
N.....
I.....
HANOI.....
N.....
P.....
Q.....
WRITELN.....
Program EXAMPLE3
  Var I : Int
  Var N : Int
  Var P : Int
  Var Q : Int
  Var C : Char
  Procedure HANOI
    Param N : Int
    Param S : Int
    Param Z : Int
```

2. All kinds of errors that can be found by the semantic analyzer, except ERR\_UNDECLARED\_INT\_CONSTANT

## 2.1. ERR\_UNDECLARED\_IDENT

Error: R:=2;

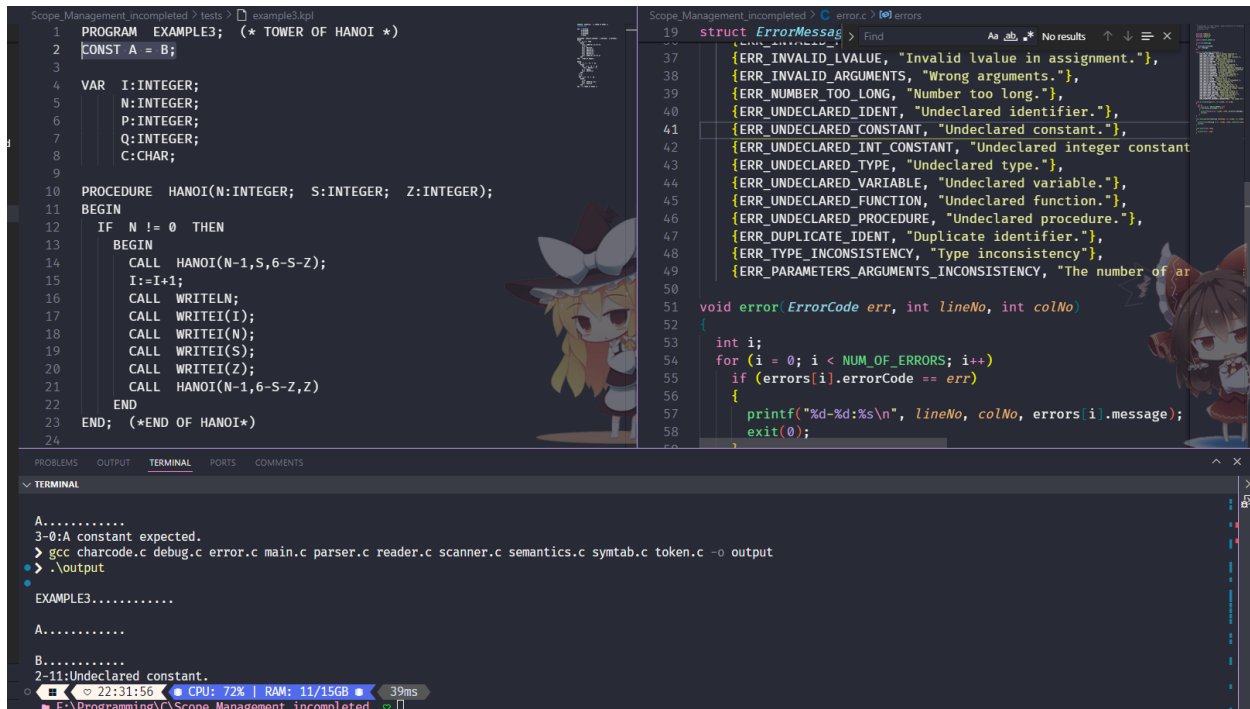
```
Scope_Management_incompleted > tests > example3.kpl
9 BEGIN
10 IF N != 0 THEN
11 BEGIN
12 END
13 END; (*END OF HANOI*)
14
15 BEGIN
16 FOR N:= 1 TO 4 DO
17 BEGIN
18 FOR I:=1 TO 4 DO
19 CALL WRITEC(' ');
20 C := READC;
21 CALL WRITEC(C)
22 END;
23 P:=1;
24 Q:=2;
25 R:=2;
26 FOR N:=2 TO 4 DO
27 BEGIN
28 I:=0;
29 CALL HANOI(N,P,Q);
30 CALL WRITELN
31 END
32 END. (* TOWER OF HANOI *)

Scope_Management_incompleted > C:\error.c > errors
19 struct ErrorMessage {
32 {ERR_INVALID_STATEMENT, "Invalid statement."},
33 {ERR_INVALID_COMPARATOR, "A comparator expected."},
34 {ERR_INVALID_EXPRESSION, "Invalid expression."},
35 {ERR_INVALID_TERM, "Invalid term."},
36 {ERR_INVALID_FACTOR, "Invalid factor."},
37 {ERR_INVALID_LVALUE, "Invalid lvalue in assignment."},
38 {ERR_INVALID_ARGUMENTS, "Wrong arguments."},
39 {ERR_NUMBER_TOO_LONG, "Number too long."},
40 {ERR_UNDECLARED_IDENT, "Undeclared identifier."},
41 {ERR_UNDECLARED_CONSTANT, "Undeclared constant."},
42 {ERR_UNDECLARED_INT_CONSTANT, "Undeclared integer constant"},
43 {ERR_UNDECLARED_TYPE, "Undeclared type."},
44 {ERR_UNDECLARED_VARIABLE, "Undeclared variable."},
45 {ERR_UNDECLARED_FUNCTION, "Undeclared function."},
46 {ERR_UNDECLARED_PROCEDURE, "Undeclared procedure."},
47 {ERR_DUPLICATE_IDENT, "Duplicate identifier."},
48 {ERR_TYPE_INCONSISTENCY, "Type inconsistency"},
49 {ERR_PARAMETERS_ARGUMENTS_INCONSISTENCY, "The number of ar
50
51 void error(ErrorCode err, int lineNo, int colNo)
52 {
53 int i;
54 for (i = 0; i < NUM_OF_ERRORS; i++)

PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
TERMINAL
READC.....
WRITEC.....
C.....
P.....
Q.....
R.....
33-3:Undeclared identifier.
22:21:43 CPU: 72% RAM: 11/15GB 50ms
```

## 2.2. ERR\_UNDECLARED\_CONSTANT

Error: CONST A = B;



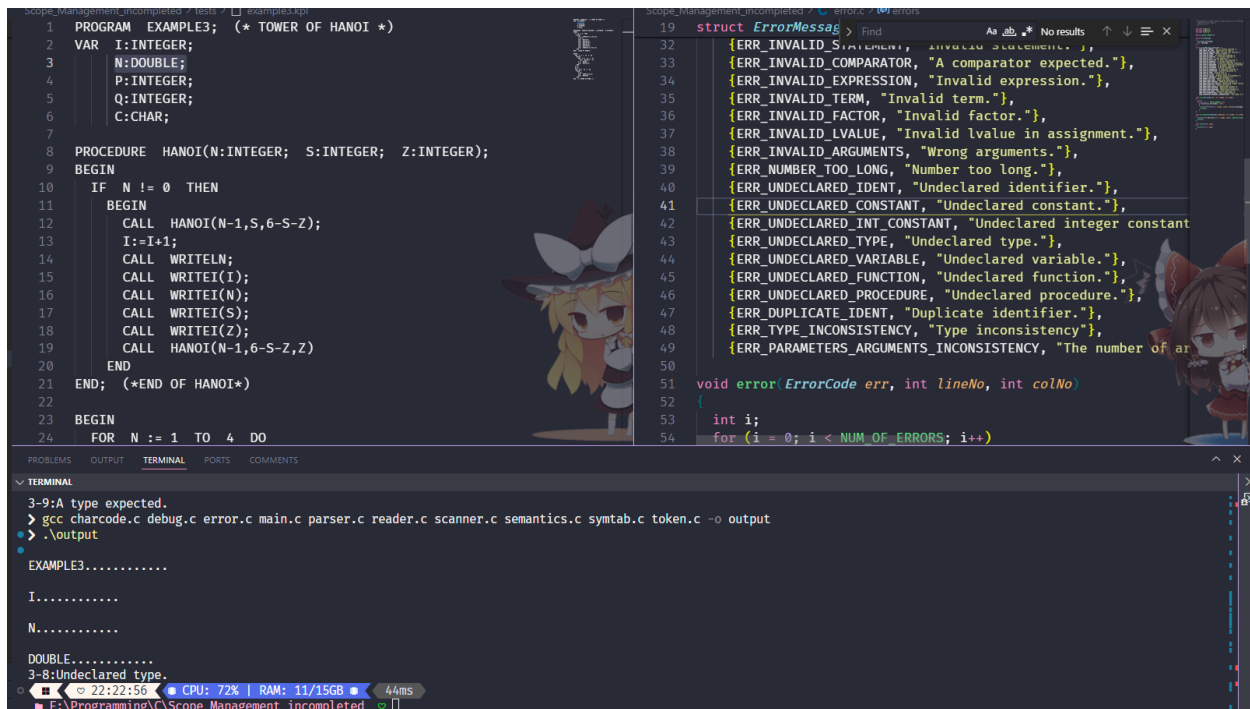
```
1 PROGRAM EXAMPLE3; (* TOWER OF HANOI *)
2 CONST A = B;
3
4 VAR I:INTEGER;
5     N:INTEGER;
6     P:INTEGER;
7     Q:INTEGER;
8     C:CHAR;
9
10 PROCEDURE HANOI(N:INTEGER; S:INTEGER; Z:INTEGER);
11 BEGIN
12     IF N != 0 THEN
13     BEGIN
14         CALL HANOI(N-1,S,6-S-Z);
15         I:=I+1;
16         CALL WRITELN;
17         CALL WRITEI(I);
18         CALL WRITEI(N);
19         CALL WRITEI(S);
20         CALL WRITEI(Z);
21         CALL HANOI(N-1,6-S-Z,Z)
22     END
23 END; (*END OF HANOI*)
24
```

```
19 struct ErrorMessage {
20     int lineNo;
21     int colNo;
22     char message[100];
23 };
24
25 void error(ErrorMessage err, int lineNo, int colNo)
26 {
27     int i;
28     for (i = 0; i < NUM_OF_ERRORS; i++)
29     {
30         if (errors[i].errorCode == err)
31         {
32             printf("%d-%d:%s\n", lineNo, colNo, errors[i].message);
33             exit(0);
34         }
35     }
36 }
```

3-0:A constant expected.  
gcc charcode.c debug.c error.c main.c parser.c reader.c scanner.c semantics.c syntab.c token.c -o output  
EXAMPLE3.....  
A.....  
B.....  
2-11:Undeclared constant.

## 2.3. ERR\_UNDECLARED\_TYPE

Error: N:DOUBLE;



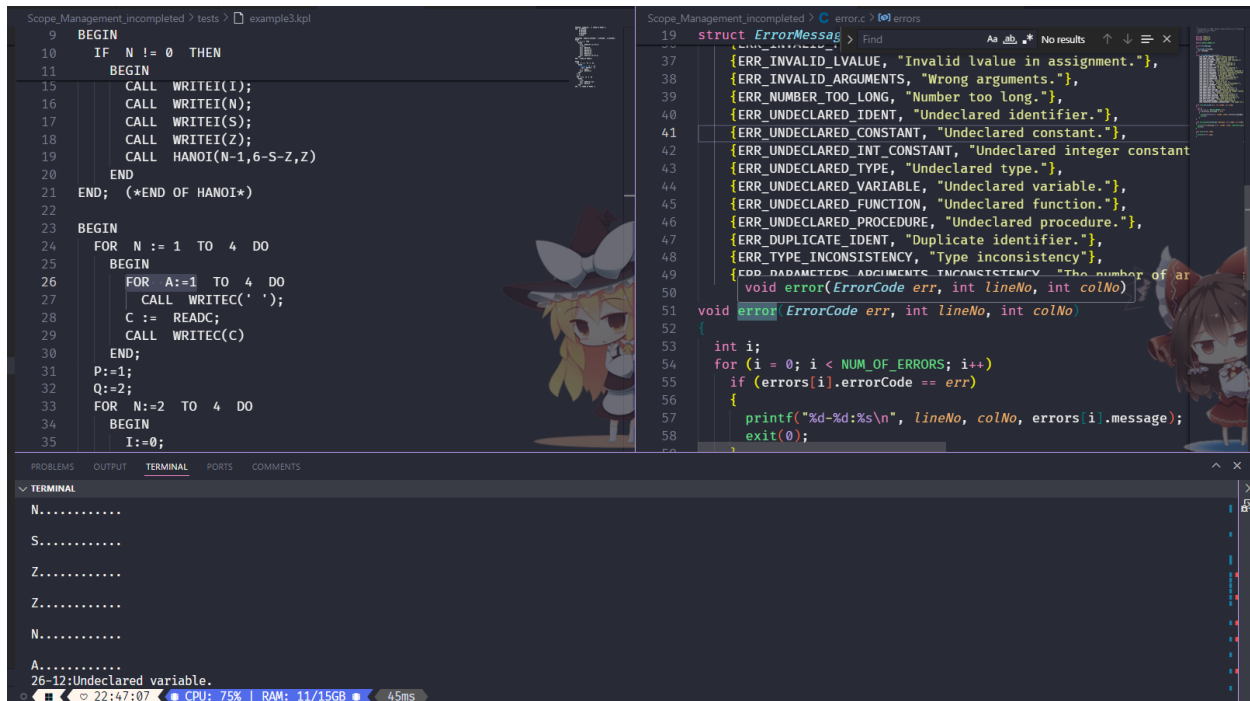
```
1 PROGRAM EXAMPLE3; (* TOWER OF HANOI *)
2 VAR I:INTEGER;
3     N:DOUBLE;
4     P:INTEGER;
5     Q:INTEGER;
6     C:CHAR;
7
8 PROCEDURE HANOI(N:INTEGER; S:INTEGER; Z:INTEGER);
9 BEGIN
10     IF N != 0 THEN
11     BEGIN
12         CALL HANOI(N-1,S,6-S-Z);
13         I:=I+1;
14         CALL WRITELN;
15         CALL WRITEI(I);
16         CALL WRITEI(N);
17         CALL WRITEI(S);
18         CALL WRITEI(Z);
19         CALL HANOI(N-1,6-S-Z,Z)
20     END
21 END; (*END OF HANOI*)
22
23 BEGIN
24     FOR N := 1 TO 4 DO
25
```

```
19 struct ErrorMessage {
20     int lineNo;
21     int colNo;
22     char message[100];
23 };
24
25 void error(ErrorMessage err, int lineNo, int colNo)
26 {
27     int i;
28     for (i = 0; i < NUM_OF_ERRORS; i++)
29     {
30         if (errors[i].errorCode == err)
31         {
32             printf("%d-%d:%s\n", lineNo, colNo, errors[i].message);
33             exit(0);
34         }
35     }
36 }
```

3-9:A type expected.  
gcc charcode.c debug.c error.c main.c parser.c reader.c scanner.c semantics.c syntab.c token.c -o output  
EXAMPLE3.....  
I.....  
N.....  
DOUBLE.....  
3-8:Undeclared type.

## 2.4. ERR\_UNDECLARED\_VARIABLE

Error: FOR A:=1



The screenshot shows a code editor with two panes. The left pane displays a Pascal program snippet. The right pane shows a list of error messages. The terminal at the bottom shows the output of the program.

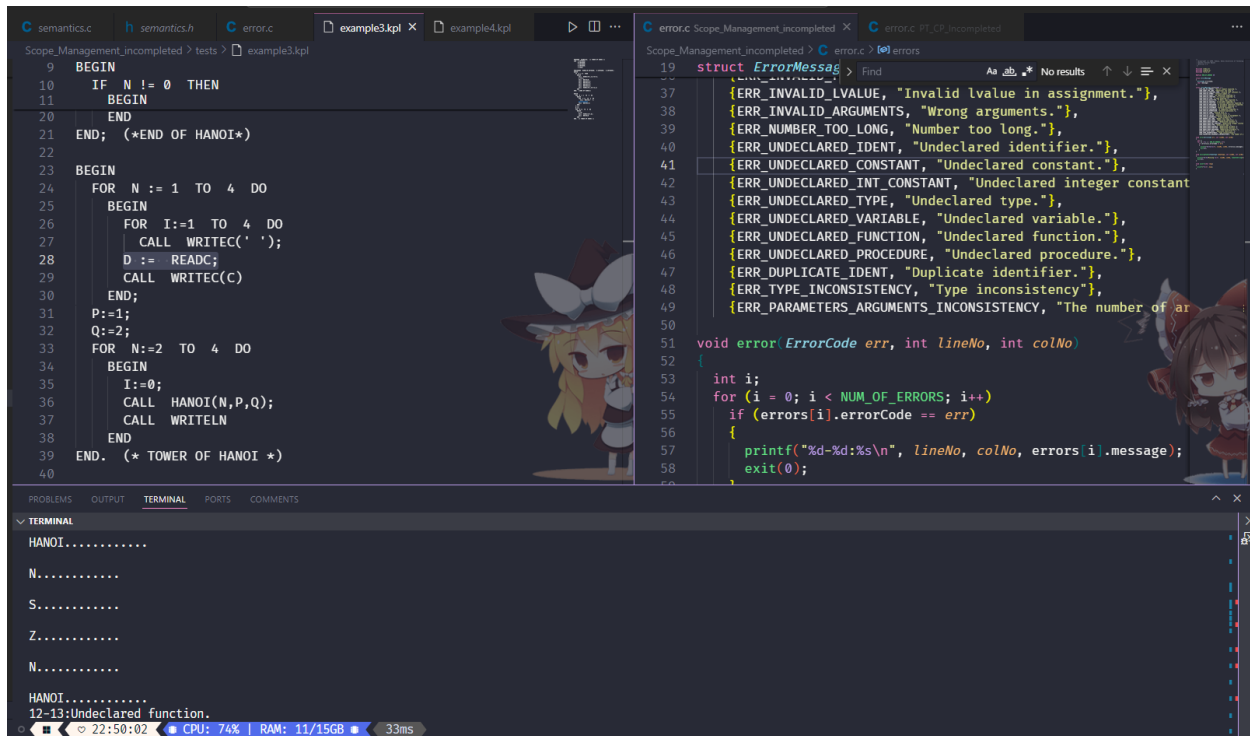
```
9 BEGIN
10 IF N != 0 THEN
11 BEGIN
15 CALL WRITEI(I);
16 CALL WRITEI(N);
17 CALL WRITEI(S);
18 CALL WRITEI(Z);
19 CALL HANOI(N-1,6-S-Z,Z)
20 END
21 END; (*END OF HANOI*)
22
23 BEGIN
24 FOR N := 1 TO 4 DO
25 BEGIN
26 FOR A:=1 TO 4 DO
27 CALL WRITEC(' ');
28 C := READC;
29 CALL WRITEC(C)
30 END;
31 P:=1;
32 Q:=2;
33 FOR N:=2 TO 4 DO
34 BEGIN
35 I:=0;
```

```
19 struct ErrorMessage
37 {ERR_INVALID_LVALUE, "Invalid lvalue in assignment."},
38 {ERR_INVALID_ARGUMENTS, "Wrong arguments."},
39 {ERR_NUMBER_TOO_LONG, "Number too long."},
40 {ERR_UNDECLARED_IDENT, "Undeclared identifier."},
41 {ERR_UNDECLARED_CONSTANT, "Undeclared constant."},
42 {ERR_UNDECLARED_INT_CONSTANT, "Undeclared integer constant"},
43 {ERR_UNDECLARED_TYPE, "Undeclared type."},
44 {ERR_UNDECLARED_VARIABLE, "Undeclared variable."},
45 {ERR_UNDECLARED_FUNCTION, "Undeclared function."},
46 {ERR_UNDECLARED_PROCEDURE, "Undeclared procedure."},
47 {ERR_DUPLICATE_IDENT, "Duplicate identifier."},
48 {ERR_TYPE_INCONSISTENCY, "Type inconsistency"},
49 {ERR_PARAMETERS_ARGUMENTS_INCONSISTENCY, "The number of ar
50 void error(ErrorCode err, int lineNo, int colNo)
51 void error(ErrorCode err, int lineNo, int colNo)
52 {
53 int i;
54 for (i = 0; i < NUM_OF_ERRORS; i++)
55 if (errors[i].errorCode == err)
56 {
57 printf("%d-%d:%s\n", lineNo, colNo, errors[i].message);
58 exit(0);
59 }
```

```
PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
TERMINAL
N.....
S.....
Z.....
Z.....
N.....
A.....
26-12:Undeclared variable.
22:47:07 CPU: 75% RAM: 11/15GB 45ms
```

## 2.5. ERR\_UNDECLARED\_FUCNTION

Error: D := READC;



The screenshot shows a code editor with two panes. The left pane displays a Pascal program snippet. The right pane shows a list of error messages. The terminal at the bottom shows the output of the program.

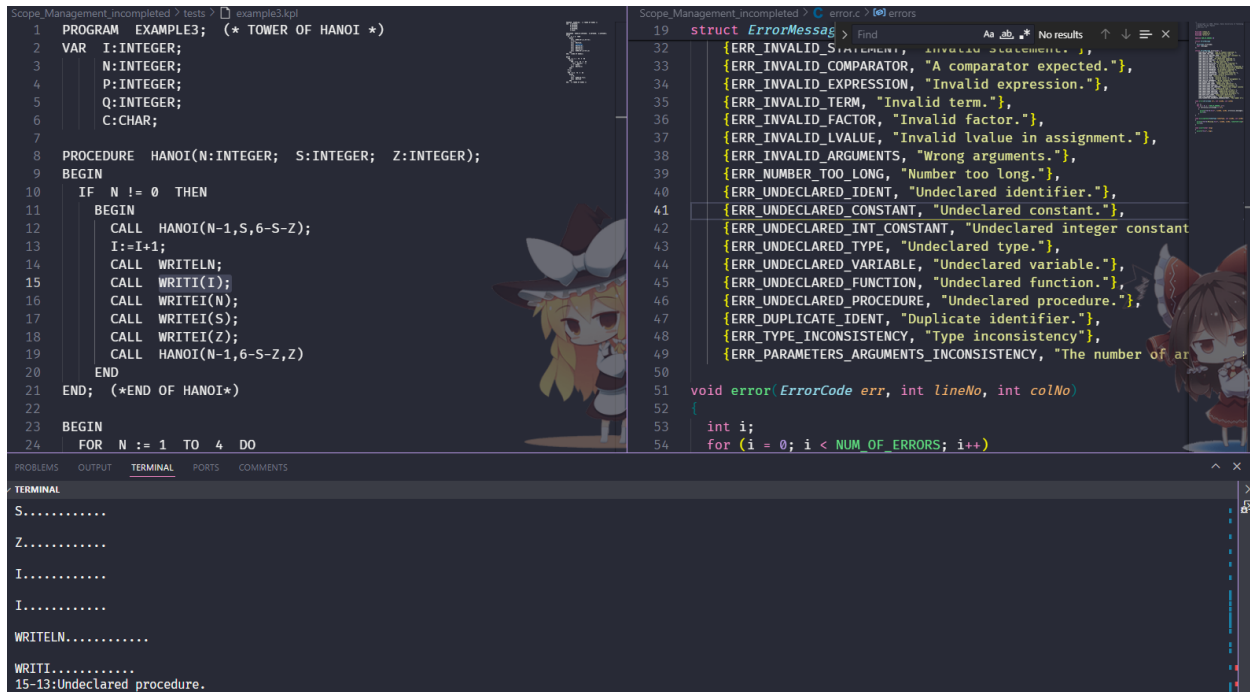
```
9 BEGIN
10 IF N != 0 THEN
11 BEGIN
20 END
21 END; (*END OF HANOI*)
22
23 BEGIN
24 FOR N := 1 TO 4 DO
25 BEGIN
26 FOR I:=1 TO 4 DO
27 CALL WRITEC(' ');
28 D := READC;
29 CALL WRITEC(C)
30 END;
31 P:=1;
32 Q:=2;
33 FOR N:=2 TO 4 DO
34 BEGIN
35 I:=0;
36 CALL HANOI(N,P,Q);
37 CALL WRITELN
38 END
39 END. (* TOWER OF HANOI *)
40
```

```
19 struct ErrorMessage
37 {ERR_INVALID_LVALUE, "Invalid lvalue in assignment."},
38 {ERR_INVALID_ARGUMENTS, "Wrong arguments."},
39 {ERR_NUMBER_TOO_LONG, "Number too long."},
40 {ERR_UNDECLARED_IDENT, "Undeclared identifier."},
41 {ERR_UNDECLARED_CONSTANT, "Undeclared constant."},
42 {ERR_UNDECLARED_INT_CONSTANT, "Undeclared integer constant"},
43 {ERR_UNDECLARED_TYPE, "Undeclared type."},
44 {ERR_UNDECLARED_VARIABLE, "Undeclared variable."},
45 {ERR_UNDECLARED_FUNCTION, "Undeclared function."},
46 {ERR_UNDECLARED_PROCEDURE, "Undeclared procedure."},
47 {ERR_DUPLICATE_IDENT, "Duplicate identifier."},
48 {ERR_TYPE_INCONSISTENCY, "Type inconsistency"},
49 {ERR_PARAMETERS_ARGUMENTS_INCONSISTENCY, "The number of ar
50 void error(ErrorCode err, int lineNo, int colNo)
51 void error(ErrorCode err, int lineNo, int colNo)
52 {
53 int i;
54 for (i = 0; i < NUM_OF_ERRORS; i++)
55 if (errors[i].errorCode == err)
56 {
57 printf("%d-%d:%s\n", lineNo, colNo, errors[i].message);
58 exit(0);
59 }
```

```
PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
TERMINAL
HANOI.....
N.....
S.....
Z.....
N.....
HANOI.....
12-13:Undeclared function.
22:50:02 CPU: 74% RAM: 11/15GB 33ms
```

## 2.6. ERR\_UNDECLARED\_PROCEDURE

Error: WRITI(I)



The screenshot shows a code editor with a Pascal program on the left and a list of error messages on the right. The program is a Tower of Hanoi solver. The error messages list various error types, with the first one being 'ERR\_UNDECLARED\_PROCEDURE' for the procedure 'WRITI(I)'. The terminal window at the bottom shows the output of the program, including the error message '15-13:Undeclared procedure.'

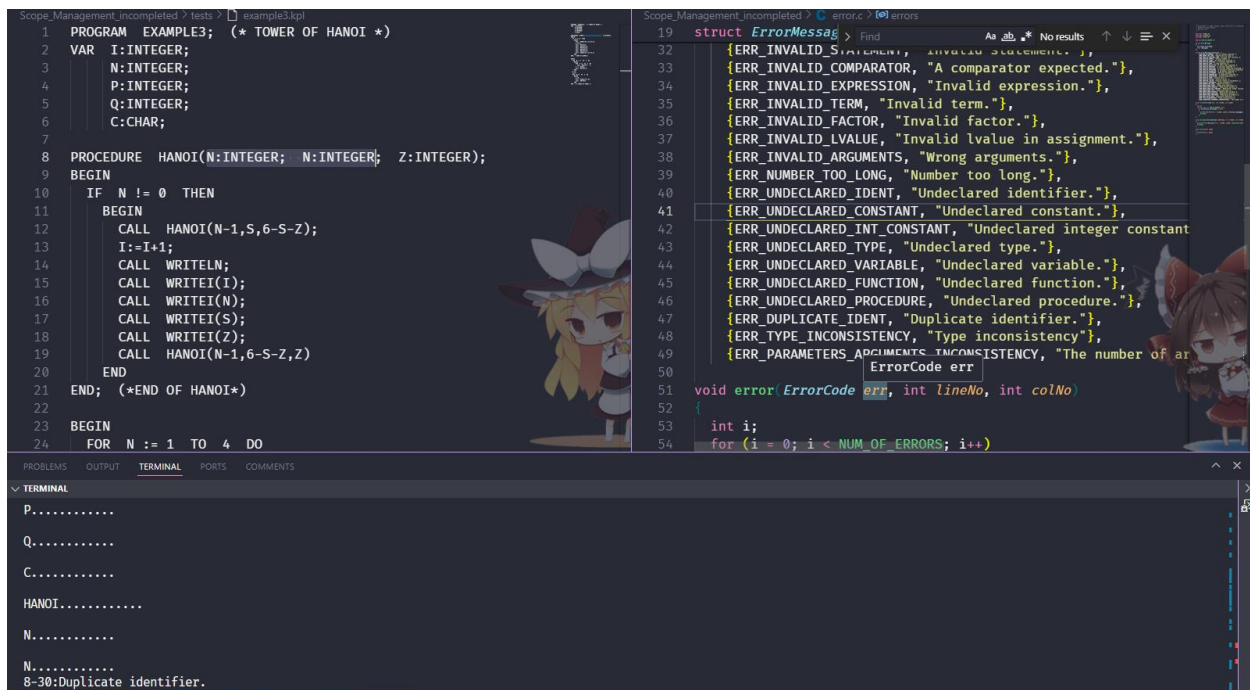
```
1 PROGRAM EXAMPLE3; (* TOWER OF HANOI *)
2 VAR I:INTEGER;
3     N:INTEGER;
4     P:INTEGER;
5     Q:INTEGER;
6     C:CHAR;
7
8 PROCEDURE HANOI(N:INTEGER; S:INTEGER; Z:INTEGER);
9 BEGIN
10  IF N != 0 THEN
11  BEGIN
12    CALL HANOI(N-1,S,6-S-Z);
13    I:=I+1;
14    CALL WRITELN;
15    CALL WRITI(I);
16    CALL WRITEI(N);
17    CALL WRITEI(S);
18    CALL WRITEI(Z);
19    CALL HANOI(N-1,6-S-Z,Z)
20  END
21 END; (*END OF HANOI*)
22
23 BEGIN
24  FOR N := 1 TO 4 DO
```

```
19 struct ErrorMessage > Find
32 {ERR_INVALID_STATEMENT, "Invalid statement."},
33 {ERR_INVALID_COMPARATOR, "A comparator expected."},
34 {ERR_INVALID_EXPRESSION, "Invalid expression."},
35 {ERR_INVALID_TERM, "Invalid term."},
36 {ERR_INVALID_FACTOR, "Invalid factor."},
37 {ERR_INVALID_LVALUE, "Invalid lvalue in assignment."},
38 {ERR_INVALID_ARGUMENTS, "Wrong arguments."},
39 {ERR_NUMBER_TOO_LONG, "Number too long."},
40 {ERR_UNDECLARED_IDENT, "Undeclared identifier."},
41 {ERR_UNDECLARED_CONSTANT, "Undeclared constant."},
42 {ERR_UNDECLARED_INT_CONSTANT, "Undeclared integer constant"},
43 {ERR_UNDECLARED_TYPE, "Undeclared type."},
44 {ERR_UNDECLARED_VARIABLE, "Undeclared variable."},
45 {ERR_UNDECLARED_FUNCTION, "Undeclared function."},
46 {ERR_UNDECLARED_PROCEDURE, "Undeclared procedure."},
47 {ERR_DUPLICATE_IDENT, "Duplicate identifier."},
48 {ERR_TYPE_INCONSISTENCY, "Type inconsistency"},
49 {ERR_PARAMETERS_ARGUMENTS_INCONSISTENCY, "The number of ar
50
51 void error(ErrorMessage err, int lineNo, int colNo)
52 {
53  int i;
54  for (i = 0; i < NUM_OF_ERRORS; i++)
```

```
PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
TERMINAL
S.....
Z.....
I.....
I.....
WRITELN.....
WRITI.....
15-13:Undeclared procedure.
```

## 2.7. ERR\_DUPLICATE\_IDENT

Error: N:INTEGER; N:INTEGER



The screenshot shows a code editor with a Pascal program on the left and a list of error messages on the right. The program is a Tower of Hanoi solver. The error messages list various error types, with the first one being 'ERR\_DUPLICATE\_IDENT' for the variable 'N'. The terminal window at the bottom shows the output of the program, including the error message '8-30:Duplicate identifier.'

```
1 PROGRAM EXAMPLE3; (* TOWER OF HANOI *)
2 VAR I:INTEGER;
3     N:INTEGER;
4     P:INTEGER;
5     Q:INTEGER;
6     C:CHAR;
7
8 PROCEDURE HANOI(N:INTEGER; N:INTEGER; Z:INTEGER);
9 BEGIN
10  IF N != 0 THEN
11  BEGIN
12    CALL HANOI(N-1,S,6-S-Z);
13    I:=I+1;
14    CALL WRITELN;
15    CALL WRITEI(I);
16    CALL WRITEI(N);
17    CALL WRITEI(S);
18    CALL WRITEI(Z);
19    CALL HANOI(N-1,6-S-Z,Z)
20  END
21 END; (*END OF HANOI*)
22
23 BEGIN
24  FOR N := 1 TO 4 DO
```

```
19 struct ErrorMessage > Find
32 {ERR_INVALID_STATEMENT, "Invalid statement."},
33 {ERR_INVALID_COMPARATOR, "A comparator expected."},
34 {ERR_INVALID_EXPRESSION, "Invalid expression."},
35 {ERR_INVALID_TERM, "Invalid term."},
36 {ERR_INVALID_FACTOR, "Invalid factor."},
37 {ERR_INVALID_LVALUE, "Invalid lvalue in assignment."},
38 {ERR_INVALID_ARGUMENTS, "Wrong arguments."},
39 {ERR_NUMBER_TOO_LONG, "Number too long."},
40 {ERR_UNDECLARED_IDENT, "Undeclared identifier."},
41 {ERR_UNDECLARED_CONSTANT, "Undeclared constant."},
42 {ERR_UNDECLARED_INT_CONSTANT, "Undeclared integer constant"},
43 {ERR_UNDECLARED_TYPE, "Undeclared type."},
44 {ERR_UNDECLARED_VARIABLE, "Undeclared variable."},
45 {ERR_UNDECLARED_FUNCTION, "Undeclared function."},
46 {ERR_UNDECLARED_PROCEDURE, "Undeclared procedure."},
47 {ERR_DUPLICATE_IDENT, "Duplicate identifier."},
48 {ERR_TYPE_INCONSISTENCY, "Type inconsistency"},
49 {ERR_PARAMETERS_ARGUMENTS_INCONSISTENCY, "The number of ar
50
51 void error(ErrorMessage err, int lineNo, int colNo)
52 {
53  int i;
54  for (i = 0; i < NUM_OF_ERRORS; i++)
```

```
PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
TERMINAL
P.....
Q.....
C.....
HANOI.....
N.....
N.....
8-30:Duplicate identifier.
```