Name: Nguyễn Hữu Hoàng Hải Anh

Student ID: 20226010

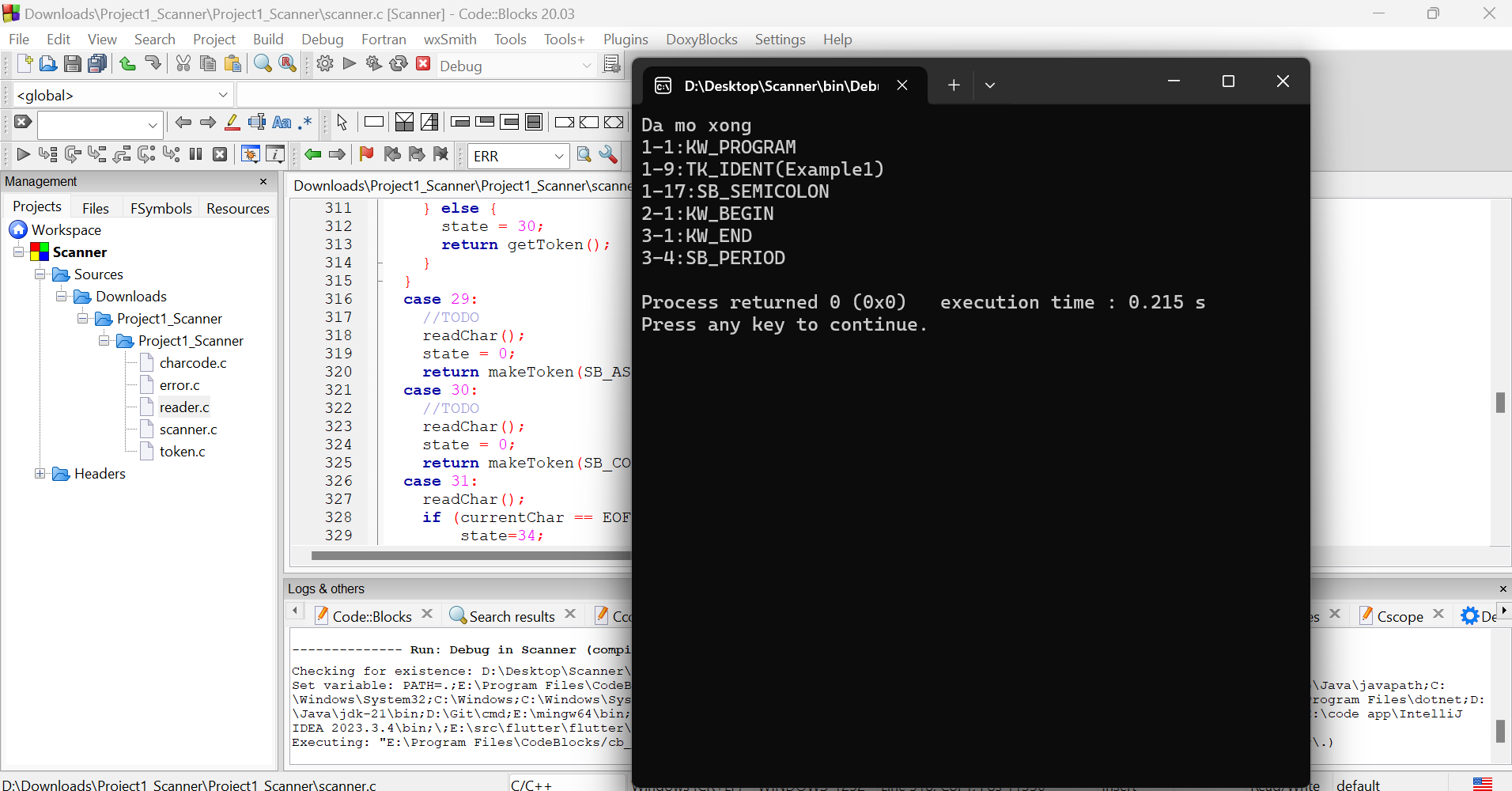
**LAB 1. SCANNER**

*This report includes*

1. Results received when executing project with files
2. All types of errors that can be found by the scanner and other KPL test files
3. 43 Cases follow the correct automata

**A, Results received when executing project with files**

1. **Example 1 (*example1.kpl*)**

****

***Results received:***

Da mo xong

1-1:KW\_PROGRAM

1-9:TK\_IDENT(Example1)

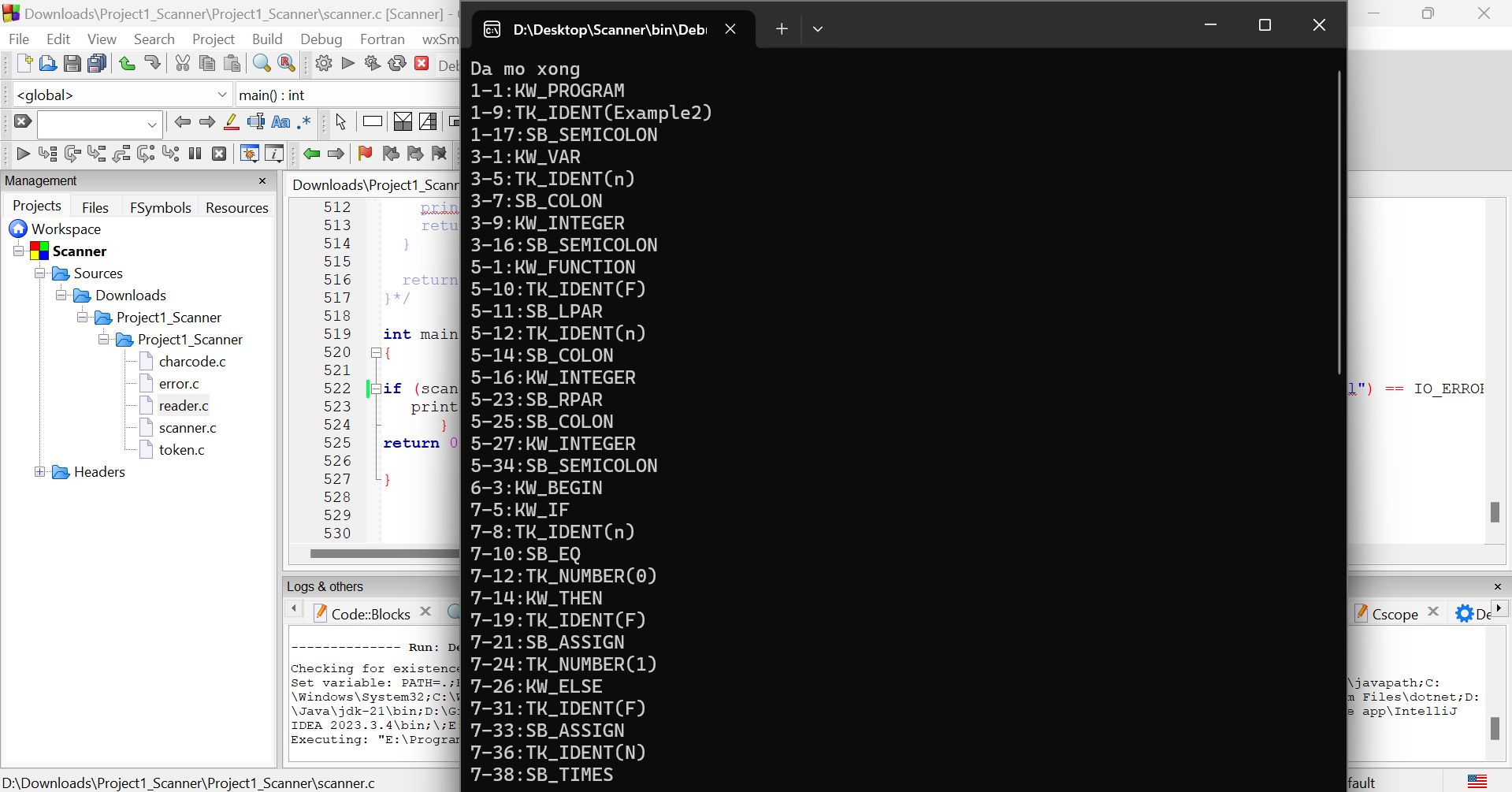
1-17:SB\_SEMICOLON

2-1:KW\_BEGIN

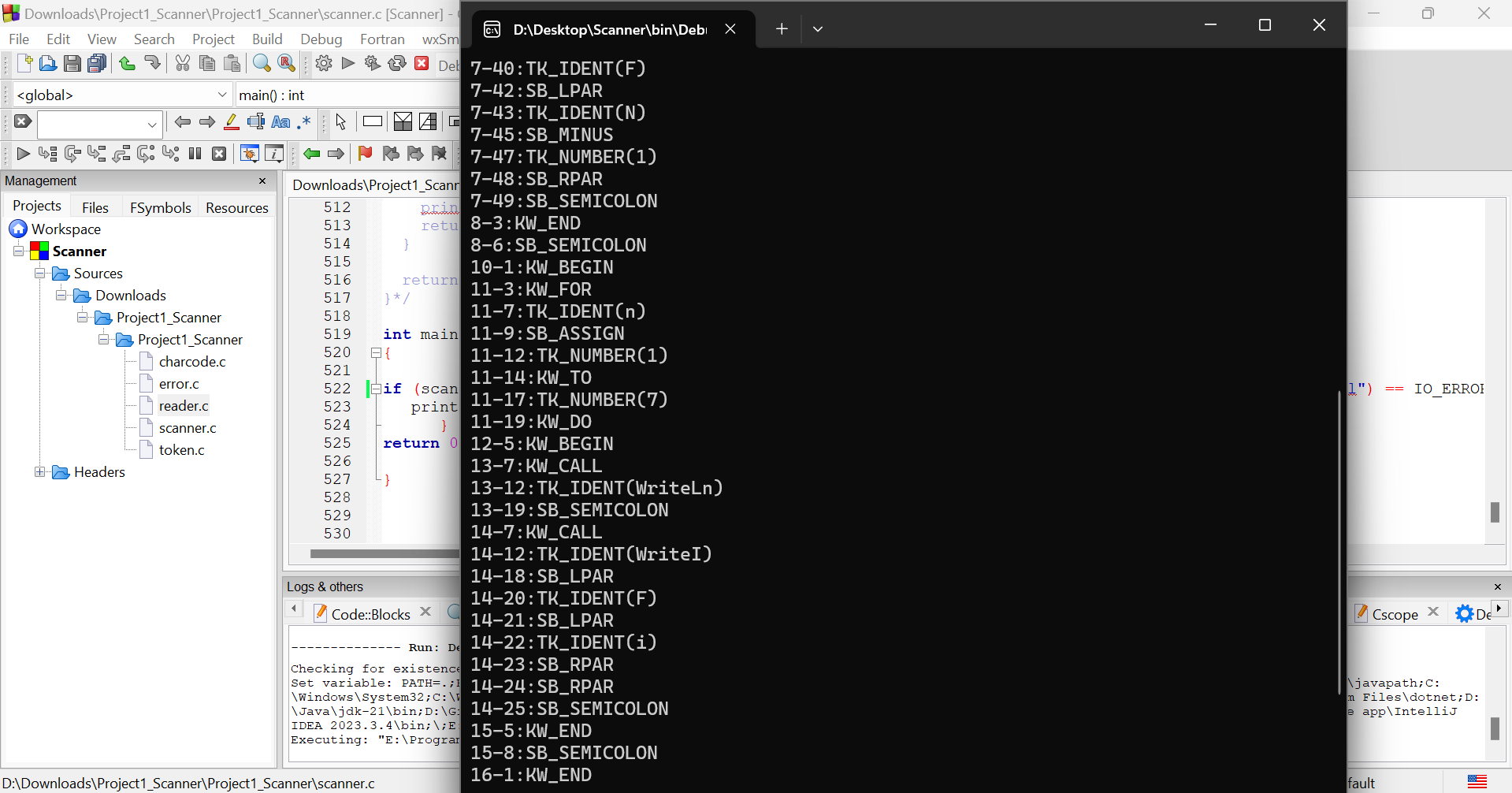
3-1:KW\_END

3-4:SB\_PERIOD

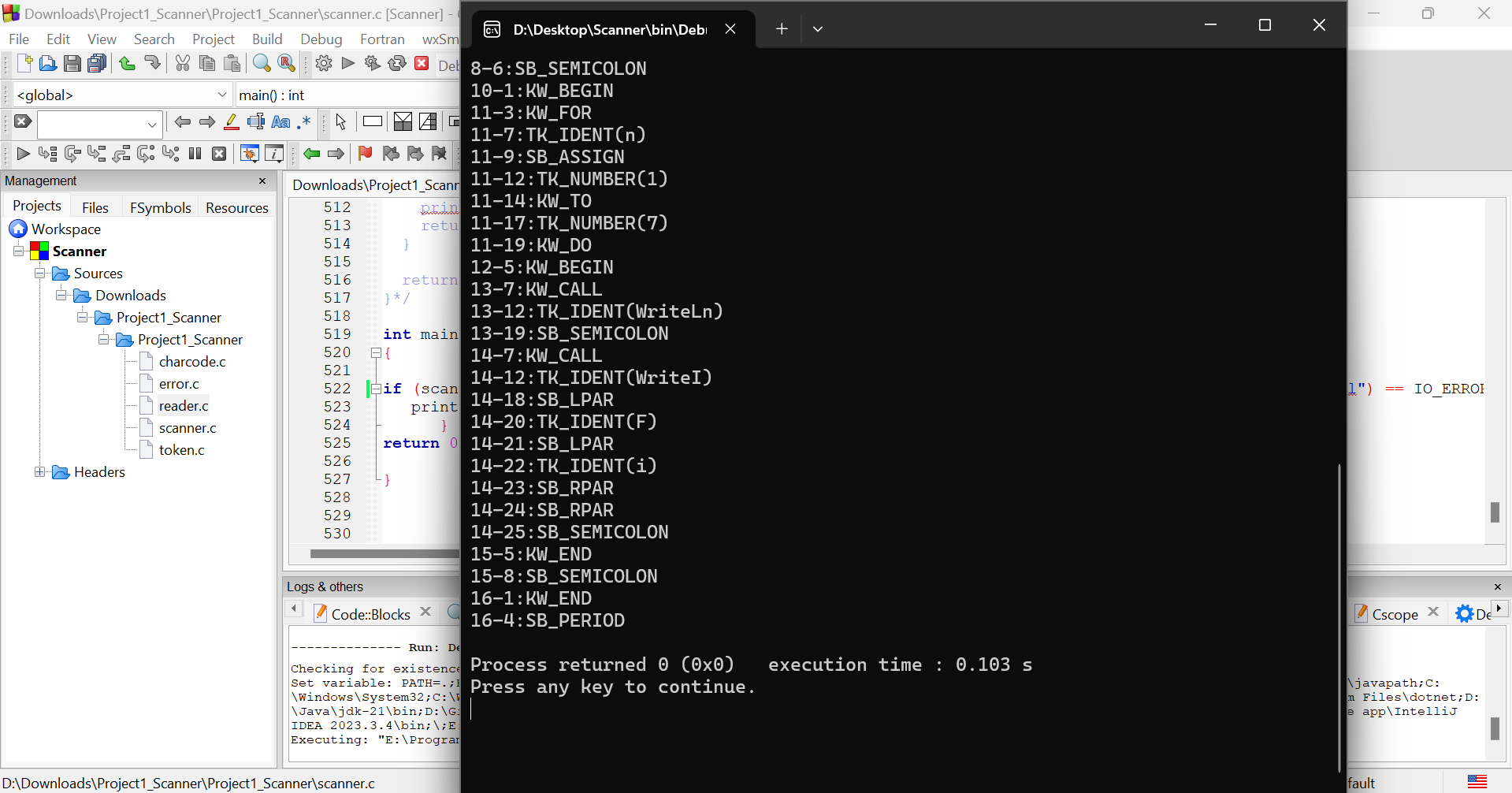
1. **Example 2 (*example2.kpl*)**

****

***Img1. Line 1 – Line 7 (38)***

****

***Img2. Line 7 (40) – Line 16 (1)***

****

***Img3. Line 16 (4)***

***Results received:***

Da mo xong

1-1:KW\_PROGRAM

1-9:TK\_IDENT(Example2)

1-17:SB\_SEMICOLON

3-1:KW\_VAR

3-5:TK\_IDENT(n)

3-7:SB\_COLON

3-9:KW\_INTEGER

3-16:SB\_SEMICOLON

5-1:KW\_FUNCTION

5-10:TK\_IDENT(F)

5-11:SB\_LPAR

5-12:TK\_IDENT(n)

5-14:SB\_COLON

5-16:KW\_INTEGER

5-23:SB\_RPAR

5-25:SB\_COLON

5-27:KW\_INTEGER

5-34:SB\_SEMICOLON

6-3:KW\_BEGIN

7-5:KW\_IF

7-8:TK\_IDENT(n)

7-10:SB\_EQ

7-12:TK\_NUMBER(0)

7-14:KW\_THEN

7-19:TK\_IDENT(F)

7-21:SB\_ASSIGN

7-24:TK\_NUMBER(1)

7-26:KW\_ELSE

7-31:TK\_IDENT(F)

7-33:SB\_ASSIGN

7-36:TK\_IDENT(N)

7-38:SB\_TIMES

7-40:TK\_IDENT(F)

7-42:SB\_LPAR

7-43:TK\_IDENT(N)

7-45:SB\_MINUS

7-47:TK\_NUMBER(1)

7-48:SB\_RPAR

7-49:SB\_SEMICOLON

8-3:KW\_END

8-6:SB\_SEMICOLON

10-1:KW\_BEGIN

11-3:KW\_FOR

11-7:TK\_IDENT(n)

11-9:SB\_ASSIGN

11-12:TK\_NUMBER(1)

11-14:KW\_TO

11-17:TK\_NUMBER(7)

11-19:KW\_DO

12-5:KW\_BEGIN

13-7:KW\_CALL

13-12:TK\_IDENT(WriteLn)

13-19:SB\_SEMICOLON

14-7:KW\_CALL

14-12:TK\_IDENT(WriteI)

14-18:SB\_LPAR

14-20:TK\_IDENT(F)

14-21:SB\_LPAR

14-22:TK\_IDENT(i)

14-23:SB\_RPAR

14-24:SB\_RPAR

14-25:SB\_SEMICOLON

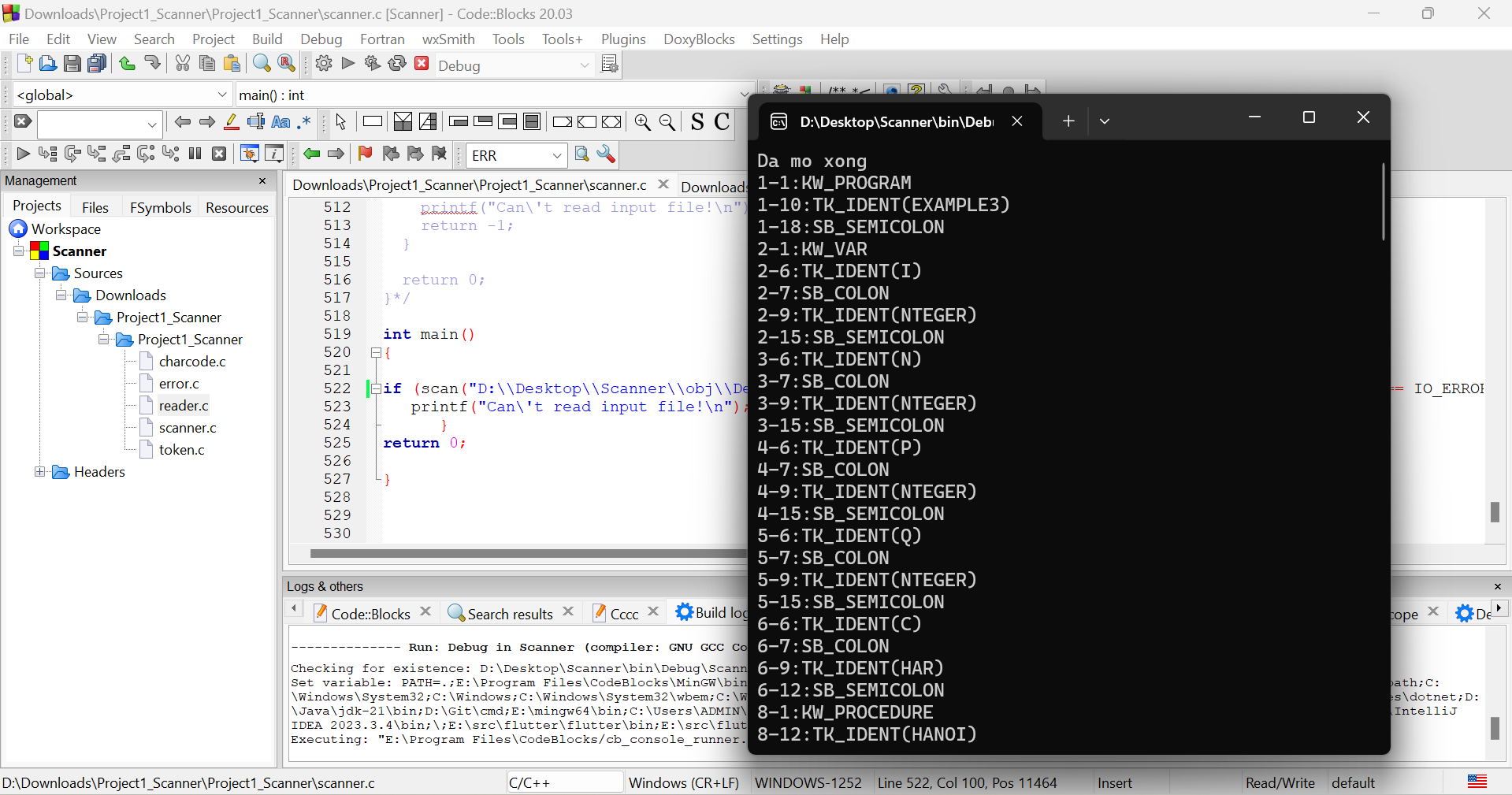
15-5:KW\_END

15-8:SB\_SEMICOLON

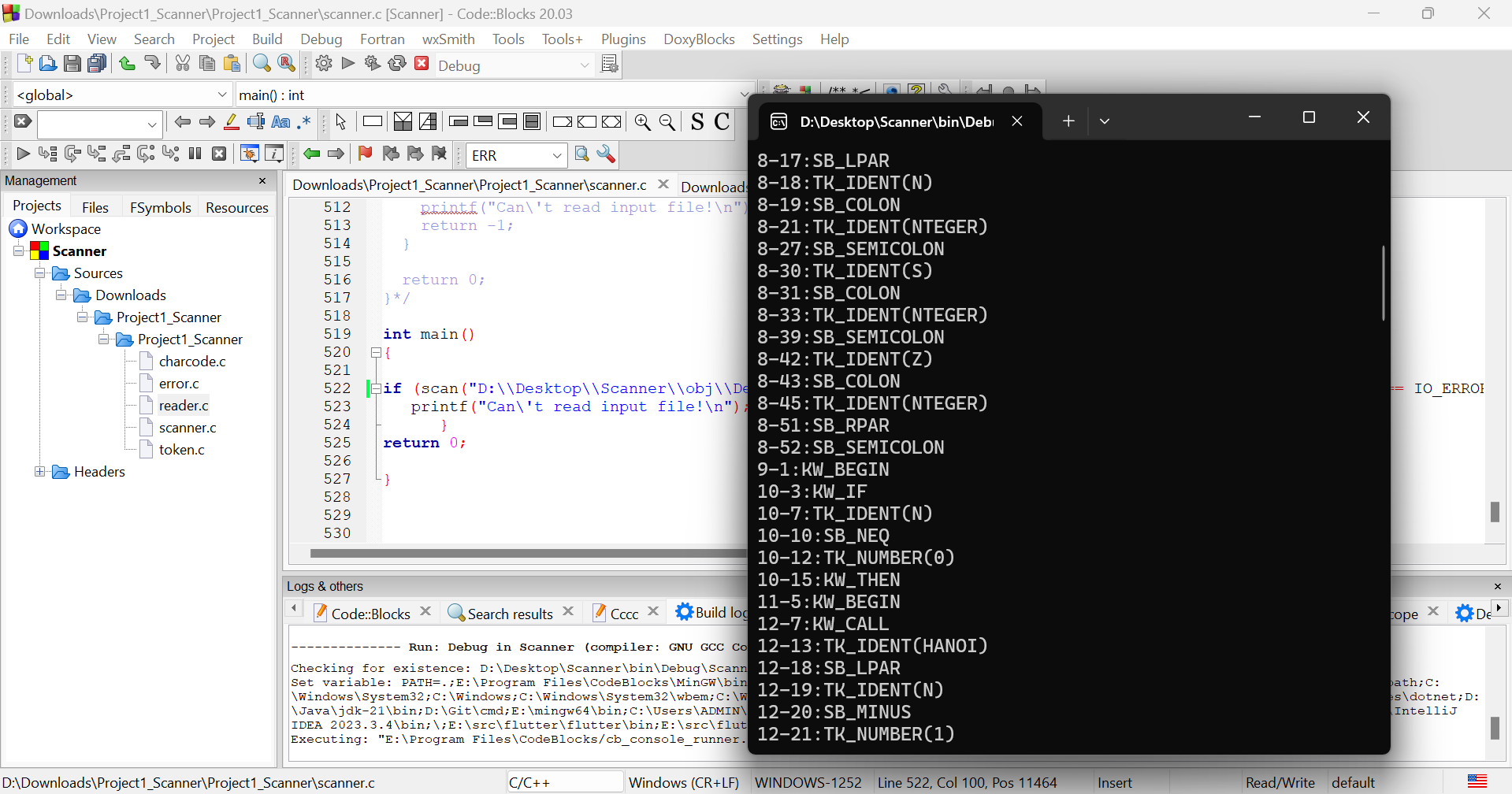
16-1:KW\_END

16-4:SB\_PERIOD

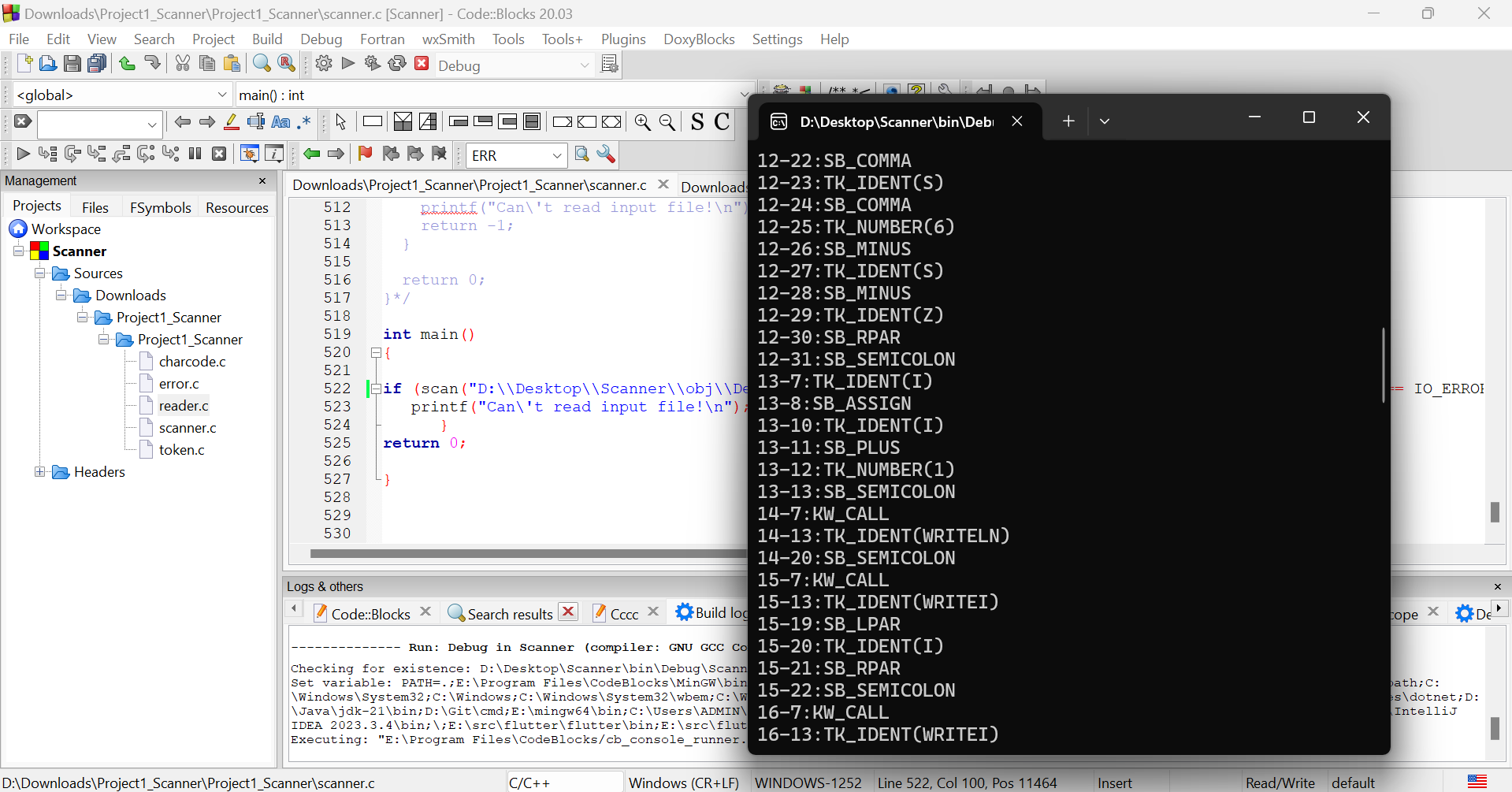
1. **Example 3 (*example3.kpl*)**

****

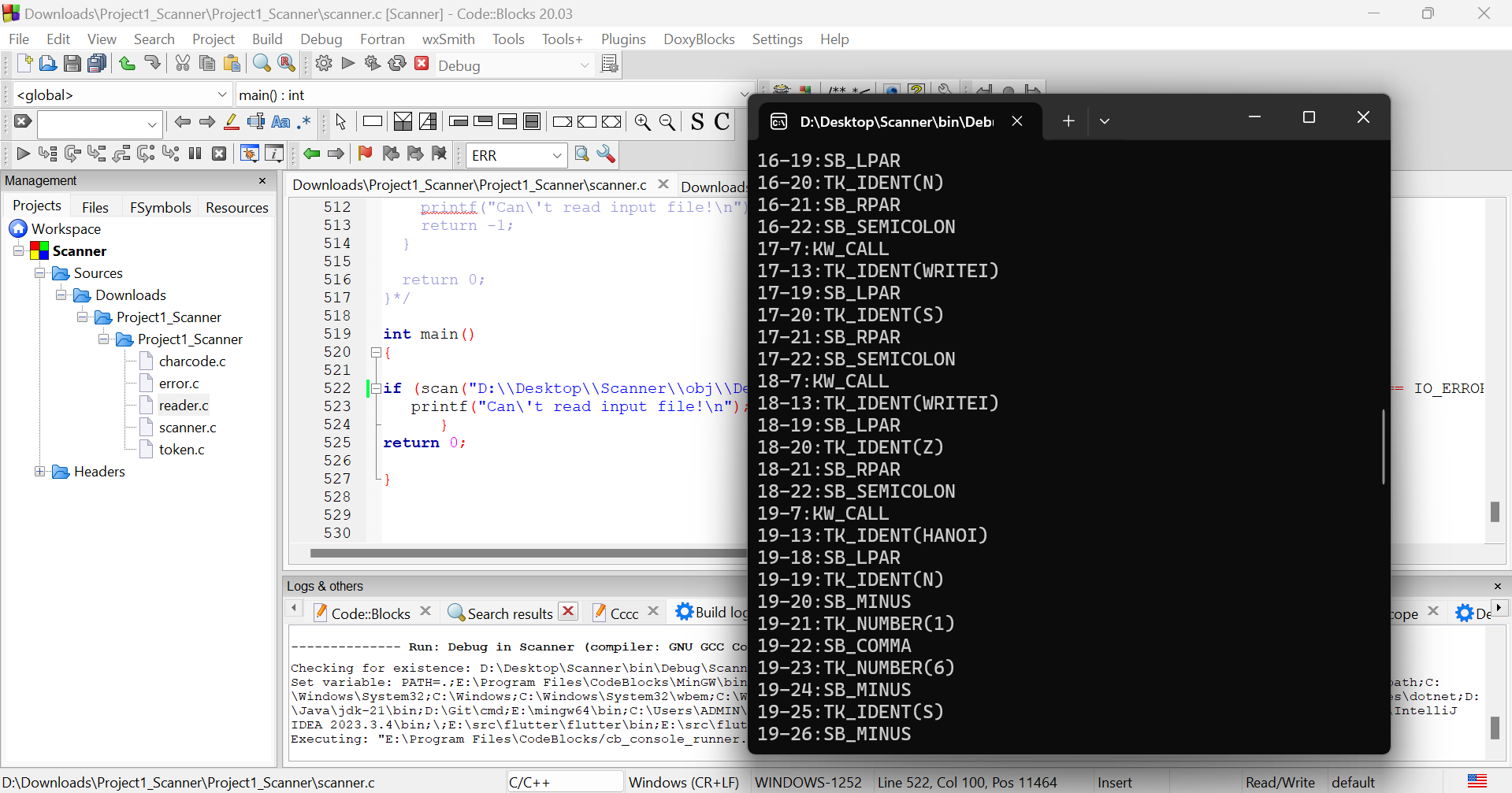
***Img4. Line 1 – Line 8 (12)***

******

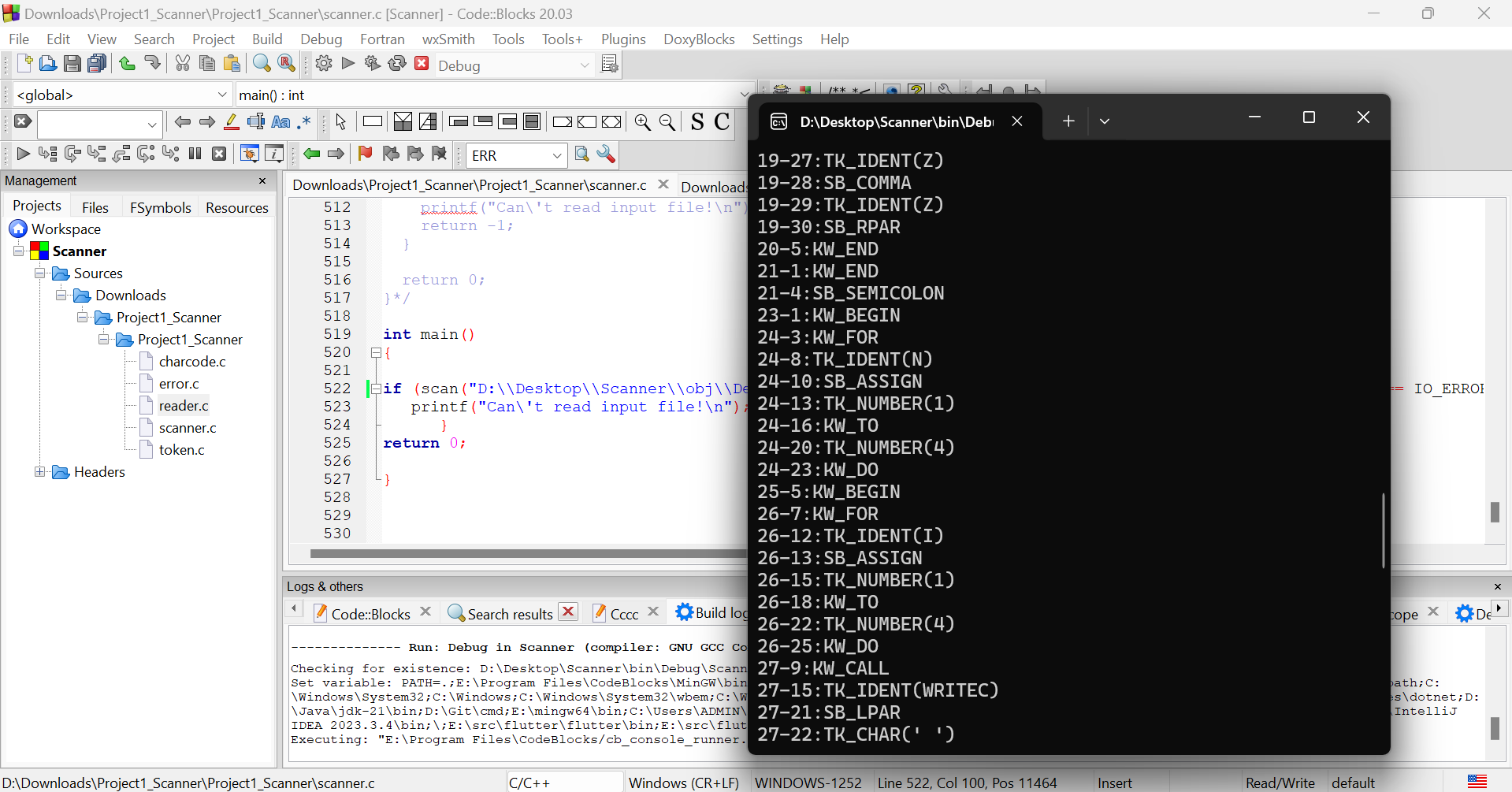
***Img5. Line 8 (17) – Line 12 (21)***

******

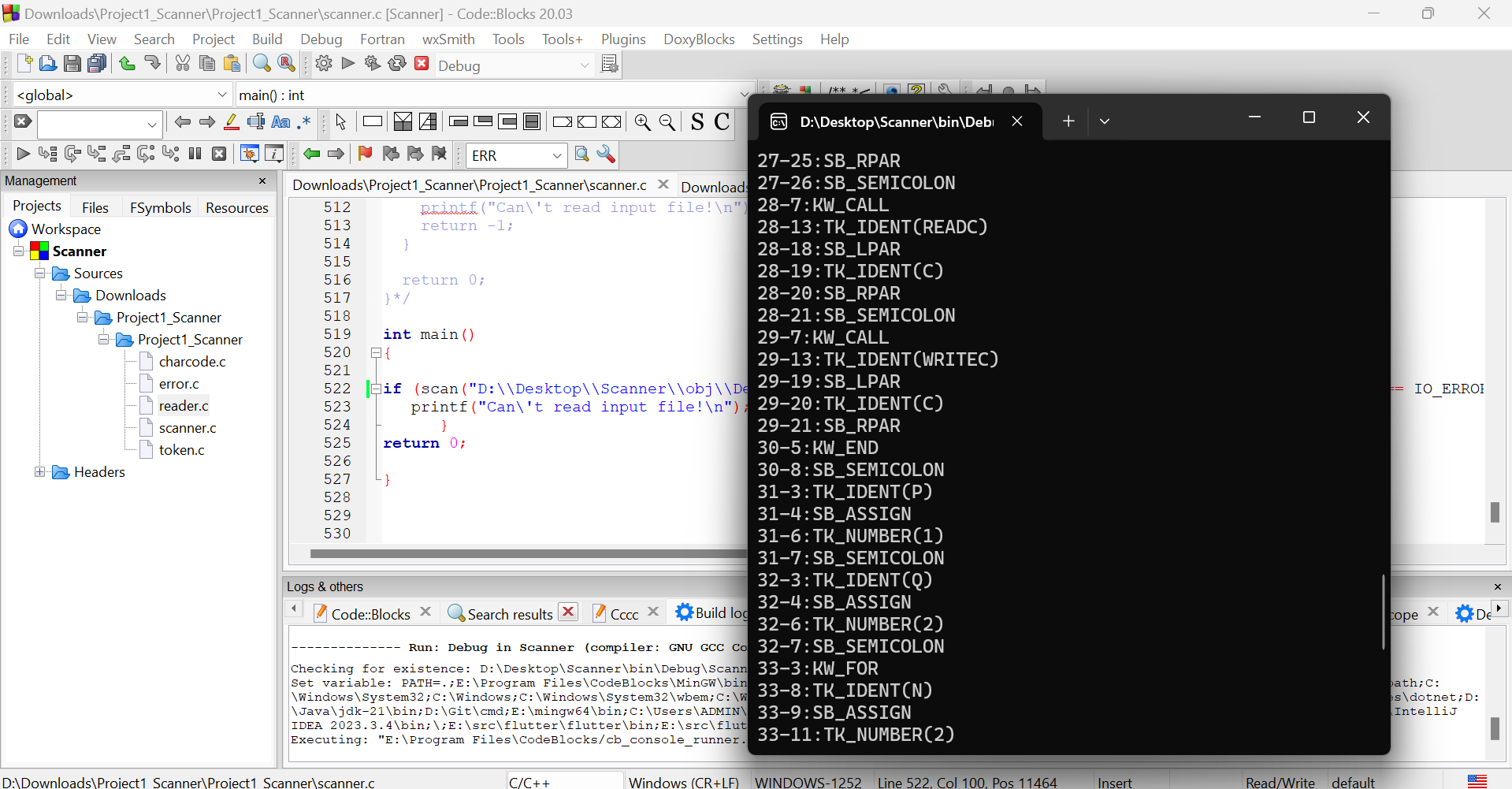
***Img6. Line 12 (22) – Line 16 (13)***

******

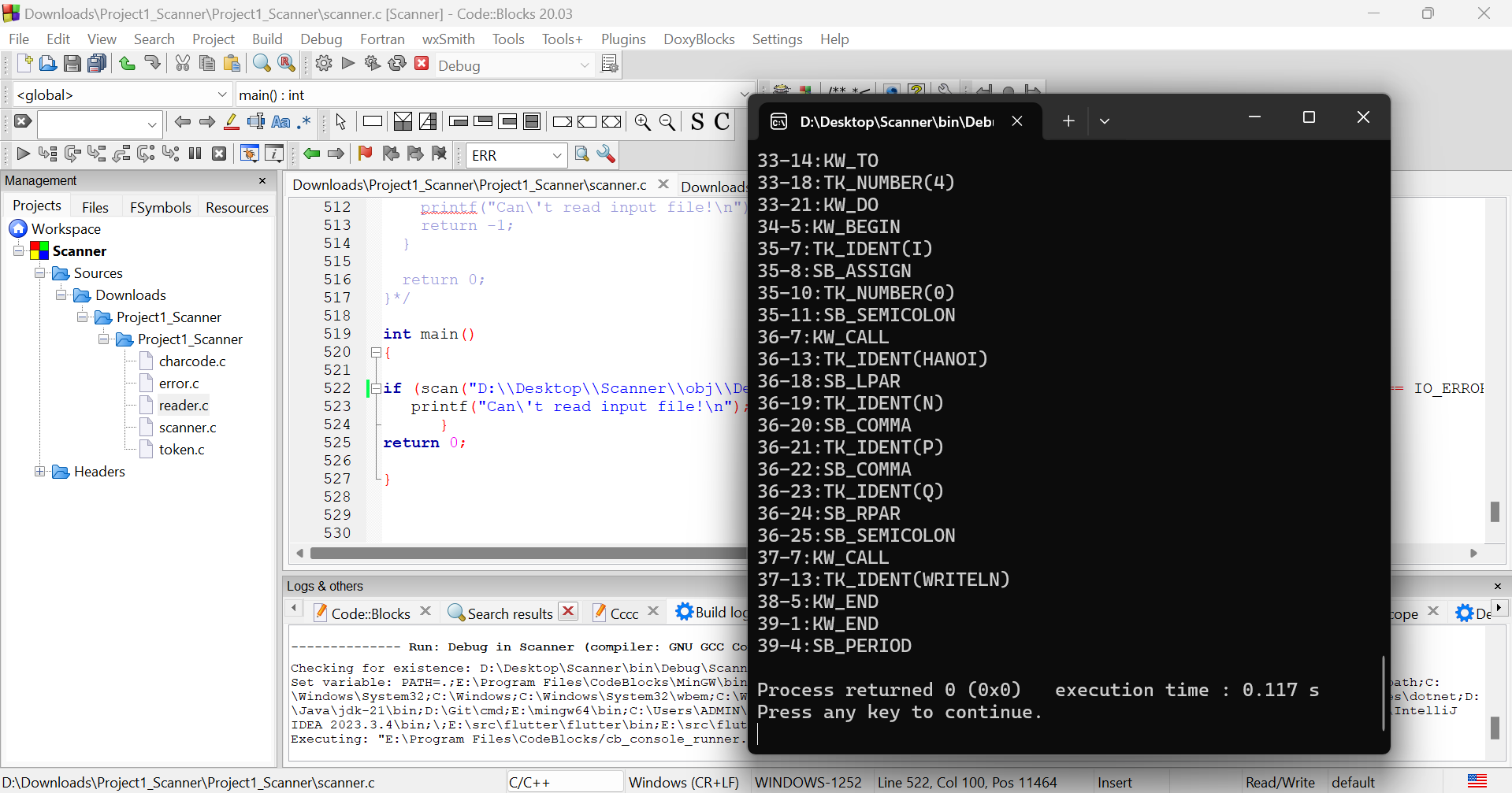
***Img7. Line 16 (19) – Line 19 (26)***

******

***Img8. Line 19 (27) – Line 27 (22)***

******

***Img9. Line 27 (25) – Line 33 (11)***

******

***Img10. Line 33 (14) – Line 39***

***Results received:***

Da mo xong

1-1:KW\_PROGRAM

1-10:TK\_IDENT(EXAMPLE3)

1-18:SB\_SEMICOLON

2-1:KW\_VAR

2-6:TK\_IDENT(I)

2-7:SB\_COLON

2-8:KW\_INTEGER

2-15:SB\_SEMICOLON

3-6:TK\_IDENT(N)

3-7:SB\_COLON

3-8:KW\_INTEGER

3-15:SB\_SEMICOLON

4-6:TK\_IDENT(P)

4-7:SB\_COLON

4-8:KW\_INTEGER

4-15:SB\_SEMICOLON

5-6:TK\_IDENT(Q)

5-7:SB\_COLON

5-8:KW\_INTEGER

5-15:SB\_SEMICOLON

6-6:TK\_IDENT(C)

6-7:SB\_COLON

6-8:KW\_CHAR

6-12:SB\_SEMICOLON

8-1:KW\_PROCEDURE

8-12:TK\_IDENT(HANOI)

8-17:SB\_LPAR

8-18:TK\_IDENT(N)

8-19:SB\_COLON

8-20:KW\_INTEGER

8-27:SB\_SEMICOLON

8-30:TK\_IDENT(S)

8-31:SB\_COLON

8-32:KW\_INTEGER

8-39:SB\_SEMICOLON

8-42:TK\_IDENT(Z)

8-43:SB\_COLON

8-44:KW\_INTEGER

8-51:SB\_RPAR

8-52:SB\_SEMICOLON

9-1:KW\_BEGIN

10-3:KW\_IF

10-7:TK\_IDENT(N)

10-9:SB\_NEQ

10-12:TK\_NUMBER(0)

10-15:KW\_THEN

11-5:KW\_BEGIN

12-7:KW\_CALL

12-13:TK\_IDENT(HANOI)

12-18:SB\_LPAR

12-19:TK\_IDENT(N)

12-20:SB\_MINUS

12-21:TK\_NUMBER(1)

12-22:SB\_COMMA

12-23:TK\_IDENT(S)

12-24:SB\_COMMA

12-25:TK\_NUMBER(6)

12-26:SB\_MINUS

12-27:TK\_IDENT(S)

12-28:SB\_MINUS

12-29:TK\_IDENT(Z)

12-30:SB\_RPAR

12-31:SB\_SEMICOLON

13-7:TK\_IDENT(I)

13-8:SB\_ASSIGN

13-10:TK\_IDENT(I)

13-11:SB\_PLUS

13-12:TK\_NUMBER(1)

13-13:SB\_SEMICOLON

14-7:KW\_CALL

14-13:TK\_IDENT(WRITELN)

14-20:SB\_SEMICOLON

15-7:KW\_CALL

15-13:TK\_IDENT(WRITEI)

15-19:SB\_LPAR

15-20:TK\_IDENT(I)

15-21:SB\_RPAR

15-22:SB\_SEMICOLON

16-7:KW\_CALL

16-13:TK\_IDENT(WRITEI)

16-19:SB\_LPAR

16-20:TK\_IDENT(N)

16-21:SB\_RPAR

16-22:SB\_SEMICOLON

17-7:KW\_CALL

17-13:TK\_IDENT(WRITEI)

17-19:SB\_LPAR

17-20:TK\_IDENT(S)

17-21:SB\_RPAR

17-22:SB\_SEMICOLON

18-7:KW\_CALL

18-13:TK\_IDENT(WRITEI)

18-19:SB\_LPAR

18-20:TK\_IDENT(Z)

18-21:SB\_RPAR

18-22:SB\_SEMICOLON

19-7:KW\_CALL

19-13:TK\_IDENT(HANOI)

19-18:SB\_LPAR

19-19:TK\_IDENT(N)

19-20:SB\_MINUS

19-21:TK\_NUMBER(1)

19-22:SB\_COMMA

19-23:TK\_NUMBER(6)

19-24:SB\_MINUS

19-25:TK\_IDENT(S)

19-26:SB\_MINUS

19-27:TK\_IDENT(Z)

19-28:SB\_COMMA

19-29:TK\_IDENT(Z)

19-30:SB\_RPAR

20-5:KW\_END

21-1:KW\_END

21-4:SB\_SEMICOLON

23-1:KW\_BEGIN

24-3:KW\_FOR

24-8:TK\_IDENT(N)

24-10:SB\_ASSIGN

24-13:TK\_NUMBER(1)

24-16:KW\_TO

24-20:TK\_NUMBER(4)

24-23:KW\_DO

25-5:KW\_BEGIN

26-7:KW\_FOR

26-12:TK\_IDENT(I)

26-13:SB\_ASSIGN

26-15:TK\_NUMBER(1)

26-18:KW\_TO

26-22:TK\_NUMBER(4)

26-25:KW\_DO

27-9:KW\_CALL

27-15:TK\_IDENT(WRITEC)

27-21:SB\_LPAR

27-22:TK\_CHAR(' ')

27-25:SB\_RPAR

27-26:SB\_SEMICOLON

28-7:KW\_CALL

28-13:TK\_IDENT(READC)

28-18:SB\_LPAR

28-19:TK\_IDENT(C)

28-20:SB\_RPAR

28-21:SB\_SEMICOLON

29-7:KW\_CALL

29-13:TK\_IDENT(WRITEC)

29-19:SB\_LPAR

29-20:TK\_IDENT(C)

29-21:SB\_RPAR

30-5:KW\_END

30-8:SB\_SEMICOLON

31-3:TK\_IDENT(P)

31-4:SB\_ASSIGN

31-6:TK\_NUMBER(1)

31-7:SB\_SEMICOLON

32-3:TK\_IDENT(Q)

32-4:SB\_ASSIGN

32-6:TK\_NUMBER(2)

32-7:SB\_SEMICOLON

33-3:KW\_FOR

33-8:TK\_IDENT(N)

33-9:SB\_ASSIGN

33-11:TK\_NUMBER(2)

33-14:KW\_TO

33-18:TK\_NUMBER(4)

33-21:KW\_DO

34-5:KW\_BEGIN

35-7:TK\_IDENT(I)

35-8:SB\_ASSIGN

35-10:TK\_NUMBER(0)

35-11:SB\_SEMICOLON

36-7:KW\_CALL

36-13:TK\_IDENT(HANOI)

36-18:SB\_LPAR

36-19:TK\_IDENT(N)

36-20:SB\_COMMA

36-21:TK\_IDENT(P)

36-22:SB\_COMMA

36-23:TK\_IDENT(Q)

36-24:SB\_RPAR

36-25:SB\_SEMICOLON

37-7:KW\_CALL

37-13:TK\_IDENT(WRITELN)

38-5:KW\_END

39-1:KW\_END

39-4:SB\_PERIOD

**B, All types of errors that can be found by the scanner and other KPL test files**

Error Handling: Scanner handles 5 unusual cases

* Identifier too long (ERR\_IDENTTOOLONG)
* Number too long (ERR\_NUMBERTOOLONG)
* Invalid character/symbol (ERR\_INVALIDSYMBOL)
* Invalid character constant (ERR\_INVALIDCHARCONSTANT)
* Comment not ended (ERR\_ENDOFCOMMENT)

*I have built a KPL program myself to detect and test all the errors defined in the lab with comments about the corresponding errors.*

PROGRAM TestErrors;

VAR

ThisIdentTooLong: INTEGER; (\* ERR\_IDENTTOOLONG \*)

A: INTEGER;

B: INTEGER;

C: INTEGER;

BEGIN

A := 12345678901; (\* ERR\_NUMBERTOOLONG \*)

B := 2;

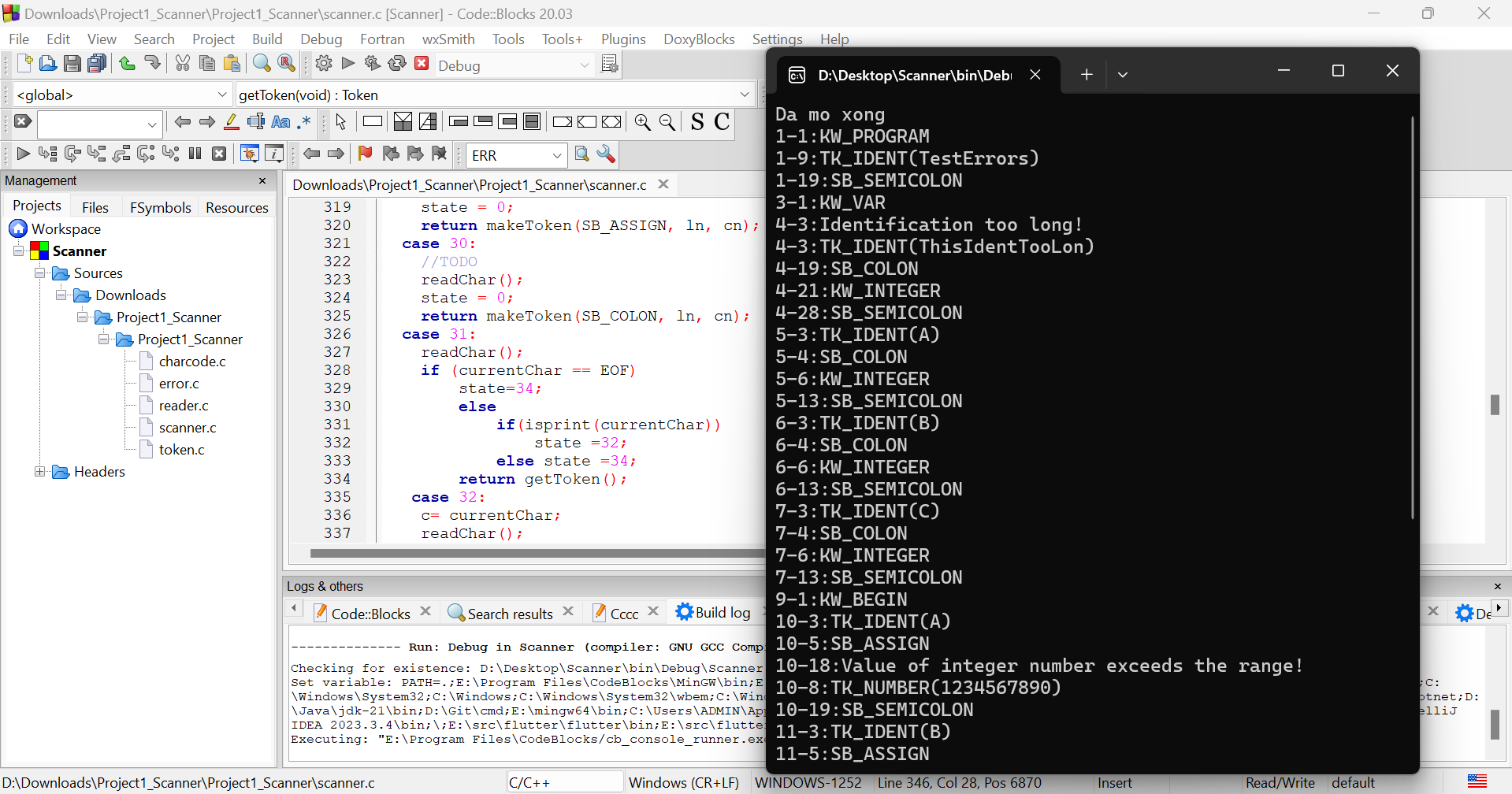
C := '&&'; (\*ERR\_INVALIDCHARCONSTANT\*)

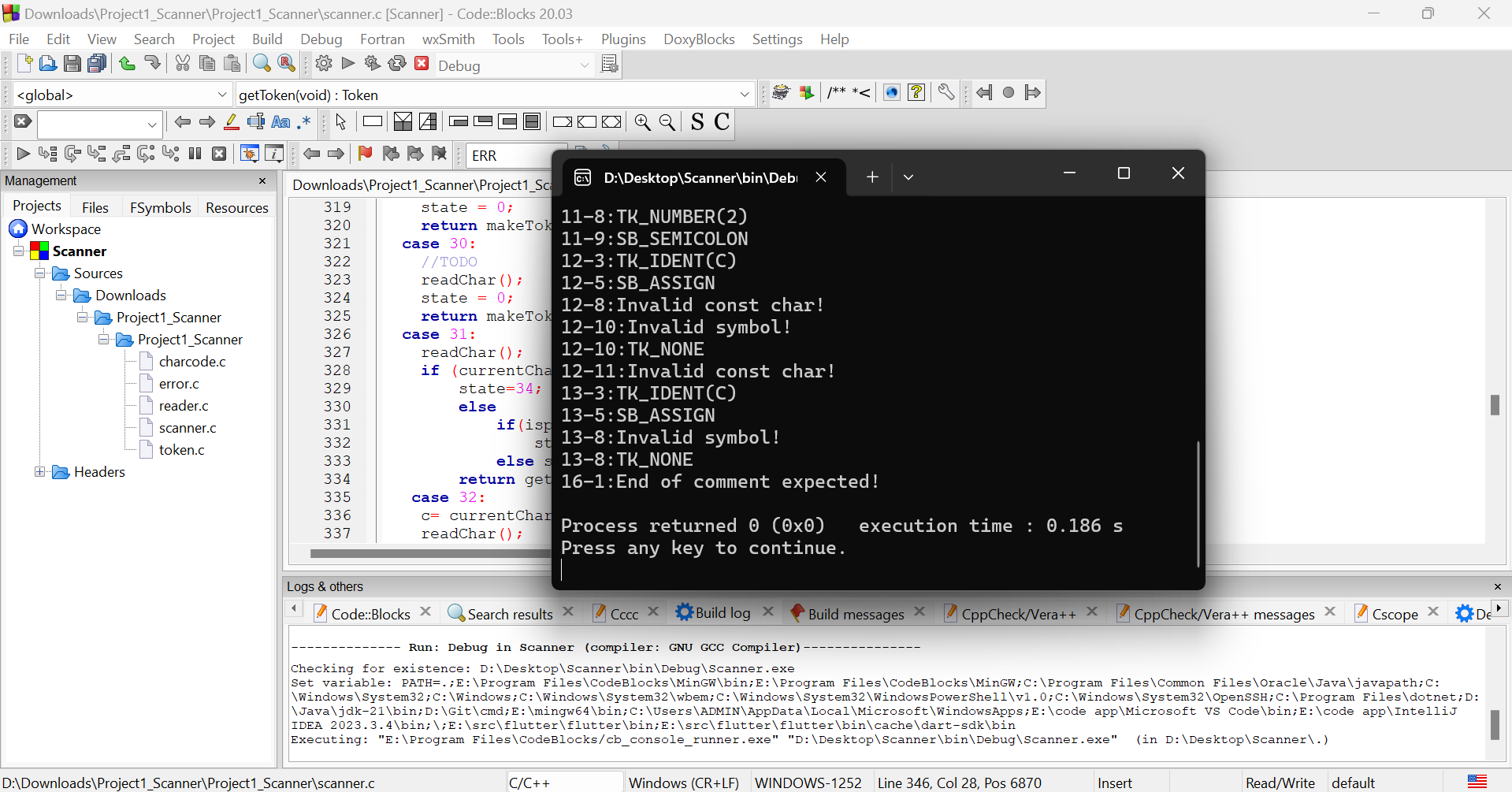
C := @ (\*ERR\_INVALIDSYMBOL\*)

(\* This is an unterminated comment --> ERR\_ENDOFCOMMENT

END.

**err1.kpl**





After executing the project with the above *err1.kpl* file, I got **the results related to the errors:**

4-3:Identification too long!

10-18:Value of integer number exceeds the range!

12-8:Invalid const char!

12-10:Invalid symbol!

12-11:Invalid const char!

13-8:Invalid symbol!

16-1:End of comment expected!

**C, 43 Cases follow the correct automata**

I have implemented 43 cases as automata in which I have added some more commands to sample cases as well as completed the //ToDo cases.

**Professors and teaching assistants can view the submission in the scanner.c file.**

For the sample cases:

* There are cases where I added the line “state = 0;” to continue identifying the rest.
* Store the starting position of each token with two variables ln (row) and cn (column) to help printToken and error know the correct position to print.
* Fix some errors to correct automata

For the //ToDo cases:

* Complete according to automata and add appropriate error detection

Add the necessary declarations

* Example: char numStr[20];

Other parts:

* I added newline characters to the reader.c file to make the printed result look nicer, for example "Da mo xong \n"