

BÁO CÁO THỰC HÀNH

XÂY DỰNG CHƯƠNG TRÌNH DỊCH

BÀI 2: PHÂN TÍCH CÚ PHÁP

Họ và tên: Bùi Quang Hưng

Mã số sinh viên: 20225849

I. Giải thích logic code

1. Kiến trúc tổng quan của Parser

Parser (bộ phân tích cú pháp) là thành phần thứ hai trong quy trình biên dịch, làm việc ngay sau Scanner. Trong khi Scanner chỉ nhận diện các token riêng lẻ, Parser có nhiệm vụ cao hơn: kiểm tra xem chuỗi các token đó có tuân theo ngữ pháp của ngôn ngữ KPL hay không.

Parser hoạt động theo phương pháp phân tích cú pháp đệ quy xuống (**Recursive Descent Parsing**) với lookahead 1 token. Điều này có nghĩa là tại mỗi thời điểm, Parser nhìn trước 1 token để quyết định nên làm gì tiếp theo.

2. Danh sách các hàm cùng chức năng trong parser

Hàm điều khiển chính

- Hàm `scan()`: Lấy token tiếp theo từ Scanner, kiểm tra token hợp lệ.
- Hàm `eat(TokenType tokenType)`: Kiểm tra token hiện tại có khớp với token mong đợi không, in token ra, lấy token tiếp theo.

Hàm phân tích chương trình

- Hàm `compileProgram()`: Parse toàn bộ chương trình KPL.
- Hàm `compileBlock()`: Parse một khối lệnh hoàn chỉnh.
- Hàm `compileBlock2()`: Parse khối lệnh không có `ConstDecls`.
- Hàm `compileBlock3()`: Parse khối lệnh không có `ConstDecls`, `TypeDecls`.
- Hàm `compileBlock4()`: Parse khối lệnh chỉ có `SubDecls` và `Statements`.
- Hàm `compileBlock5()`: Parse khối lệnh chỉ có `Statements`.

Hàm phân tích khai báo

- Hàm `compileConstDecls()`: Parse danh sách khai báo hằng.
- Hàm `compileConstDecl()`: Parse một khai báo hằng.
- Hàm `compileConstant()`: Parse hằng số có dấu.

- Hàm `compileConstant2()`: Parse phần constant sau dấu +/-.
- Hàm `compileUnsignedConstant()`: Parse hằng số không dấu.
- Hàm `compileTypeDecls()`: Parse danh sách khai báo kiểu.
- Hàm `compileTypeDecl()`: Parse một khai báo kiểu.
- Hàm `compileType()`: Parse kiểu dữ liệu.
- Hàm `compileBasicType()`: Parse kiểu cơ bản.
- Hàm `compileVarDecls()`: Parse danh sách khai báo biến.
- Hàm `compileVarDecl()`: Parse một khai báo biến.
- Hàm `compileSubDecls()`: Parse danh sách function/procedure.
- Hàm `compileFuncDecl()`: Parse khai báo function.
- Hàm `compileProcDecl()`: Parse khai báo procedure.
- Hàm `compileParams()`: Parse danh sách tham số.
- Hàm `compileParams2()`: Parse các tham số tiếp theo.
- Hàm `compileParam()`: Parse một tham số.

Hàm phân tích câu lệnh

- Hàm `compileStatements()`: Parse khối BEGIN...END.
- Hàm `compileStatements2()`: Parse các statement tiếp theo.
- Hàm `compileStatement()`: Parse một statement, phân loại theo token đầu.
- Hàm `compileAssignSt()`: Parse lệnh gán.
- Hàm `compileCallSt()`: Parse lệnh gọi procedure.
- Hàm `compileGroupSt()`: Parse nhóm lệnh.
- Hàm `compileIfSt()`: Parse câu lệnh IF.
- Hàm `compileElseSt()`: Parse phần ELSE (nếu có).
- Hàm `compileWhileSt()`: Parse vòng lặp WHILE.
- Hàm `compileForSt()`: Parse vòng lặp FOR.

Hàm phân tích biểu thức

- Hàm `compileCondition()`: Parse điều kiện.
- Hàm `compileCondition2()`: Parse toán tử so sánh và biểu thức thứ 2.
- Hàm `compileExpression()`: Parse biểu thức, xử lý dấu +/- đầu.
- Hàm `compileExpression2()`: Parse các term tiếp theo với +/-.
- Hàm `compileExpression3()`: Parse expression sau dấu +/-.
- Hàm `compileTerm()`: Parse term (phép nhân/chia).
- Hàm `compileTerm2()`: Parse các factor tiếp theo với */÷.
- Hàm `compileFactor()`: Parse factor (số, biến, hàm, biểu thức con).

Hàm phụ trợ

- Hàm `compileArguments()`: Parse danh sách đối số khi gọi hàm.
- Hàm `compileArguments2()`: Parse các đối số tiếp theo.
- Hàm `compileIndexes()`: Parse chỉ số mảng (có thể nhiều chiều).

II. Thực hiện chương trình với tệp `example1.kpl` (không lỗi)

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIncompleted> .\parser .\Test\example1.kpl
Parsing a Program ....
1-1:KW_PROGRAM
1-9:TK_IDENT(Example1)
1-17:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Subroutines parsed ....
2-1:KW_BEGIN
3-1:KW_END
Block parsed!
3-5:SB_PERIOD
Program parsed!

```

- So sánh với kết quả được cung cấp, ta thấy output trùng khớp

III. Thực hiện chương trình với tệp lỗi

- **Lỗi `ERR_END_OF_COMMENT`**

```

error1.kpl x h parser.h error8.kpl
Test > error1.kpl
1 (* Test ERR_END_OF_COMMENT - comment không đóng
2 Program Error1;
3 Begin
4 End.
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIncompleted> .\parser .\Test\example1.kpl
Parsing a Program ....
1-1:KW_PROGRAM
1-9:TK_IDENT(Example1)
1-17:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Subroutines parsed ....
2-1:KW_BEGIN
3-1:KW_END
Block parsed!
3-5:SB_PERIOD
Program parsed!
● PS D:\CTD_Lab\PT_CPIncompleted> .\parser .\Test\error1.kpl
5-1:End of comment expected!
○ PS D:\CTD_Lab\PT_CPIncompleted>

```

- **Lỗi `ERR_IDENT_TOO_LONG`**

```
Test > error2.kpl
1  (* Test ERR_IDENT_TOO_LONG - identifier quá dài >15 ký tự *)
2  Program Error2;
3  Var thisIsAVeryLongIdentifierNameThatExceeds : Integer;
4  Begin
5  End.
6

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIncompleted> .\parser .\Test\error2.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error2)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:Identification too long!
```

- Lỗi ERR_NUMBER_TOO_LONG

```
Test > error3.kpl
1  (* Test ERR_NUMBER_TOO_LONG - số quá lớn *)
2  Program Error3;
3  Const c = 999999999999;
4  Begin
5  End.
6

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIncompleted> .\parser .\Test\error3.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error3)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_CONST
3-7:TK_IDENT(c)
3-9:SB_EQ
3-11:Value of integer number exceeds the range!
○ PS D:\CTD_Lab\PT_CPIncompleted>
```

- Lỗi ERR_INVALIDCHARCONSTANT

```
Test > error4.kpl
1 (* Test ERR_INVALIDCHARCONSTANT - char constant không hợp lệ *)
2 Program Error4;
3 Var c : Char;
4 Begin
5   c := 'abc';
6 End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error4.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error4)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(c)
3-7:SB_COLON
3-9:KW_CHAR
3-13:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(c)
5-6:SB_ASSIGN
5-8:Invalid constant!
○ PS D:\CTD_Lab\PT_CPIIncompleted>
```

- Lỗi ERR_INVALID_SYMBOL

```
Test > error5.kpl
1 (* Test ERR_INVALID_SYMBOL - ký tự không hợp lệ *)
2 Program Error5;
3 Var x : Integer;
4 Begin
5   x := 10 @ 5;
6 End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error5.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error5)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:KW_INTEGER
3-16:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(x)
5-6:SB_ASSIGN
Parsing an expression
5-8:TK_NUMBER(10)
5-11:Invalid symbol!
○ PS D:\CTD_Lab\PT_CPIIncompleted>
```

- Lỗi ERR_INVALID_CONSTANT

```
Test > error6.kpl
1  (* Test ERR_INVALID_CONSTANT - constant không hợp lệ *)
2  Program Error6;
3  Var c : Char;
4  Begin
5  |  c := '';
6  End.
7
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS QUERY RESULTS

```
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error6.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error6)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(c)
3-7:SB_COLON
3-9:KW_CHAR
3-13:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(c)
5-6:SB_ASSIGN
5-8:Invalid constant!
PS D:\CTD_Lab\PT_CPIIncompleted>
```

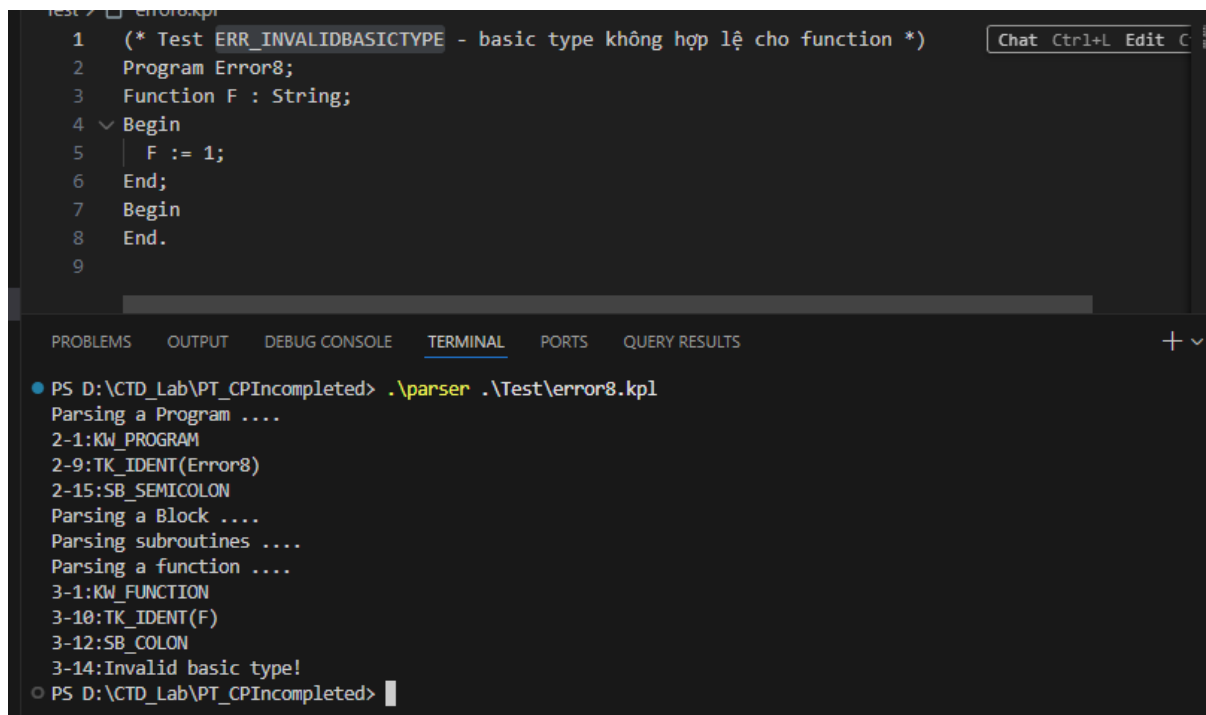
- Lỗi ERR_INVALIDTYPE

```
1  (* Test ERR_INVALIDTYPE - kiểu dữ liệu không hợp lệ *)
2  Program Error7;
3  Var x : String;
4  Begin
5  End.
6
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS QUERY RESULTS

```
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error7.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error7)
2-15:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:TK_IDENT(String)
3-15:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
5-1:KW_END
Block parsed!
6-0:SB_PERIOD
Program parsed!
```

- Lỗi ERR_INVALIDBASICTYPE



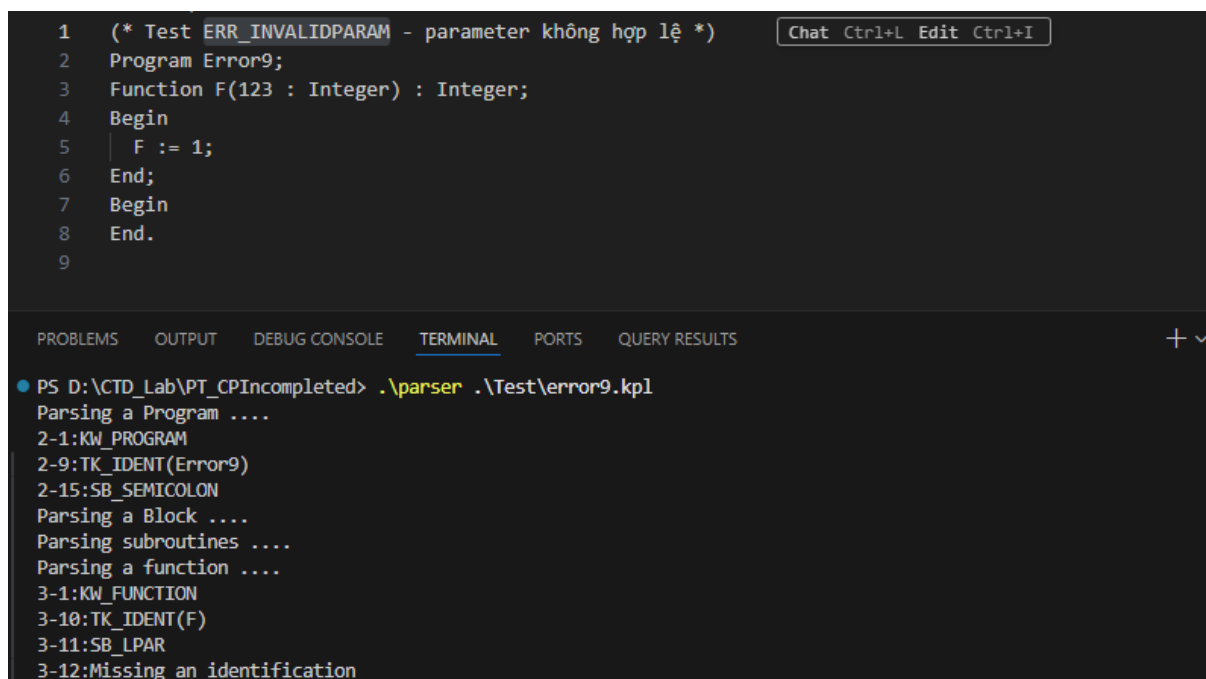
The screenshot shows a code editor with a Pascal program and its terminal output. The program is as follows:

```
1 (* Test ERR_INVALIDBASICTYPE - basic type không hợp lệ cho function *)
2 Program Error8;
3 Function F : String;
4 Begin
5   F := 1;
6 End;
7 Begin
8 End.
```

The terminal output shows the following:

```
PS D:\CTD_Lab\PT_CPI\incompleted> .\parser .\Test\error8.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error8)
2-15:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Parsing a function ....
3-1:KW_FUNCTION
3-10:TK_IDENT(F)
3-12:SB_COLON
3-14:Invalid basic type!
PS D:\CTD_Lab\PT_CPI\incompleted>
```

- Lỗi ERR_INVALIDPARAM



The screenshot shows a code editor with a Pascal program and its terminal output. The program is as follows:

```
1 (* Test ERR_INVALIDPARAM - parameter không hợp lệ *)
2 Program Error9;
3 Function F(123 : Integer) : Integer;
4 Begin
5   F := 1;
6 End;
7 Begin
8 End.
```

The terminal output shows the following:

```
PS D:\CTD_Lab\PT_CPI\incompleted> .\parser .\Test\error9.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error9)
2-15:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Parsing a function ....
3-1:KW_FUNCTION
3-10:TK_IDENT(F)
3-11:SB_LPAR
3-12:Missing an identification
```

- Lỗi ERR_INVALIDSTATEMENT

```
Test > error10.kpl
1  (* Test ERR_INVALIDSTATEMENT - statement không hợp lệ *)
2  Program Error10;
3  Begin
4  | 123;
5  End.
6

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error10.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error10)
2-16:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Subroutines parsed ....
3-1:KW_BEGIN
4-3:Invalid statement!
○ PS D:\CTD_Lab\PT_CPIIncompleted> 
```

- Lỗi ERR_INVALIDARGUMENTS

```
1  (* Test ERR_INVALIDARGUMENTS - arguments không hợp lệ *)
2  Program Error11;
3  Begin
4  | Call WriteI(@);
5  End.
6

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error11.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error11)
2-16:SB_SEMICOLON
Parsing a Block ....
Parsing subroutines ....
Subroutines parsed ....
3-1:KW_BEGIN
Parsing a call statement ....
4-3:KW_CALL
4-8:TK_IDENT(WriteI)
4-14:SB_LPAR
4-15:Invalid symbol!
```

- Lỗi ERR_INVALIDCOMPARATOR


```
Test > error12.kpl
1  (* Test ERR_INVALIDCOMPARATOR - comparator không hợp lệ *)
2  Program Error12;
3  Var x : Integer;
4  Begin
5    If x := 0 Then x := 1;
6  End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error12.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error12)
2-16:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:KW_INTEGER
3-16:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an if statement ....
5-3:KW_IF
Parsing an expression
5-6:TK_IDENT(x)
5-9:Invalid term!
○ PS D:\CTD_Lab\PT_CPIIncompleted>
```

- Lỗi ERR_INVALIDEXPRESSION

```
Test > error13.kpl
1  (* Test ERR_INVALIDEXPRESSION - expression không hợp lệ *)
2  Program Error13;
3  Var x : Integer;
4  Begin
5    x := Begin;
6  End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS
● PS D:\CTD_Lab\PT_CPIIncompleted> .\parser .\Test\error13.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error13)
2-16:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:KW_INTEGER
3-16:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(x)
5-6:SB_ASSIGN
Parsing an expression
5-8:Invalid factor!
○ PS D:\CTD_Lab\PT_CPIIncompleted>
```

- Lỗi ERR_INVALIDTERM

```
Test > error14.kpl
1  (* Test ERR_INVALIDTERM - term không hợp lệ *)
2  Program Error14;
3  Var x : Integer;
4  Begin
5    x := 10 * Begin;
6  End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

● PS D:\CTD_Lab\PT_CPI\incompleted> .\parser .\Test\error14.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error14)
2-16:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:KW_INTEGER
3-16:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(x)
5-6:SB_ASSIGN
Parsing an expression
5-8:TK_NUMBER(10)
5-11:SB_TIMES
5-13:Invalid factor!
○ PS D:\CTD_Lab\PT_CPI\incompleted>
```

- Lỗi ERR_INVALIDFACTOR

```
Test > error15.kpl
1  (* Test ERR_INVALIDFACTOR - factor không hợp lệ *)
2  Program Error15;
3  Var x : Integer;
4  Begin
5    x := * 5;
6  End.
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

● PS D:\CTD_Lab\PT_CPI\incompleted> .\parser .\Test\error15.kpl
Parsing a Program ....
2-1:KW_PROGRAM
2-9:TK_IDENT(Error15)
2-16:SB_SEMICOLON
Parsing a Block ....
3-1:KW_VAR
3-5:TK_IDENT(x)
3-7:SB_COLON
3-9:KW_INTEGER
3-16:SB_SEMICOLON
Parsing subroutines ....
Subroutines parsed ....
4-1:KW_BEGIN
Parsing an assign statement ....
5-3:TK_IDENT(x)
5-6:SB_ASSIGN
Parsing an expression
5-8:Invalid factor!
○ PS D:\CTD_Lab\PT_CPI\incompleted>
```