```
java
```

```
class ComplexErrorTest {
    // Tipo faltante en una declaración
    missing_type variable = 10;

    // Función con error en los parámetros
    void errorFunction(int a, float, string c) {
        // Falta operando en una expresión
        int result = a + * c;

        // Error en estructura de control (if sin condición)
        if () {
            print("Error");
        }

        // Estructura anidada con múltiples errores
        for (int i = 0 i < 10; i++) {
            // Errores en expresiones anidadas
            if (i > && i < 8) {
                switch (i) {
                    case 2
                        print("Two");
                    break;
                    case 4:
                        // Error en la llamada a función (falta un parámetro)
                        calculateValue(;
                    break;
                    default:
                        print("Other");

                }
            }
        }
    }

    // Error en el retorno (tipo incorrecto)
    int getValue() {
        // Debería devolver un entero pero devuelve un string
        return "value";
    }

    // Error en declaración de arreglo
    float[] getArray(int size) {
        float values[] = new float[];  // Falta el tamaño
```

```
        for (int i = 0; i < size i++) {
            values[i] = i * 2.5;
        }
        return values;
    }
}
```

Salida esperada para inputComplexErrors.txt

```
-- class
--- IDENTIFIER: ComplexErrorTest
---- {
Syntax Error 4 at line 3, token: missing_type
Recovered: Expected type keyword (int, float, etc.), found 'missing_type'
----- RULE_TYPE
----- TYPE: void
----- IDENTIFIER: errorFunction
----- (
------ RULE_PARAMS
----- RULE_TYPE
----- TYPE: int
------ IDENTIFIER: a
------ ,
----- RULE_TYPE
----- TYPE: float
Syntax Error 13 at line 6, token: ,
Recovered: Missing parameter name after type 'float'
------ ,
----- RULE_TYPE
----- TYPE: string
------ IDENTIFIER: c
----- )
----- {
-- RULE_BODY
--- RULE_VARIABLE
----- RULE_TYPE
----- TYPE: int
--- IDENTIFIER: result
--- =
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: a
------- +
Syntax Error 60 at line 8, token: *
Recovered: Expected expression operand after '+', found '*'
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: c
-- ;
```

```
--- RULE_IF
--- if
--- (
Syntax Error 1401 at line 11, token: )
Recovered: Missing condition in if statement
--- )
---- RULE_STATEMENT_BLOCK
---- {
-- RULE_BODY
--- RULE_CALL_METHOD
--- IDENTIFIER: print
--- (
---- RULE_PARAM_VALUES
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: "Error"
--- )
-- ;
---- }

--- RULE_FOR
--- for
--- (
--- RULE_VARIABLE
----- RULE_TYPE
----- TYPE: int
--- IDENTIFIER: i
--- =
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 0
Syntax Error 40 at line 16, token: i
Recovered: Missing semicolon after initialization in for loop
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ <
```

```
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 10
-- ;
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ ++
--- )
---- RULE_STATEMENT_BLOCK
---- {
-- RULE_BODY
--- RULE_IF
--- if
--- (
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ >
Syntax Error 1420 at line 18, token: &&
Recovered: Missing operand after '>' operator
------ &&
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ <
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 8
--- )
---- RULE_STATEMENT_BLOCK
---- {
-- RULE_BODY
--- RULE_SWITCH
--- switch
--- (
```

```
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
--- )
--- {
---- case
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 2
Syntax Error 45 at line 20, token: print
Recovered: Missing ':' after case expression
---- :
-- RULE_BODY
--- RULE_CALL_METHOD
--- IDENTIFIER: print
--- (
---- RULE_PARAM_VALUES
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: "Two"
--- )
-- ;
---- break
---- ;
---- case
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 4
---- :
-- RULE_BODY
--- RULE_CALL_METHOD
--- IDENTIFIER: calculateValue
--- (
Syntax Error 25 at line 23, token: ;
Recovered: Missing parameter value in function call
---- RULE_PARAM_VALUES
```

```
--- )
-- ;
---- break
---- ;
---- default
---- :
-- RULE_BODY
--- RULE_CALL_METHOD
--- IDENTIFIER: print
--- (
---- RULE_PARAM_VALUES
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: "Other"
--- )
-- ;
--- }
---- }
---- }
----- }

----- RULE_METHODS
----- RULE_TYPE
----- TYPE: int
----- IDENTIFIER: getValue
----- (
------ RULE_PARAMS
----- )
----- {
-- RULE_BODY
--- RULE_RETURN
--- return
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: "value"
Warning: Return value type mismatch. Expected int, found STRING.
-- ;
----- }

----- RULE_METHODS
```

```
----- RULE_TYPE
----- TYPE: float
--- IDENTIFIER: []
--- IDENTIFIER: getArray
----- (
------ RULE_PARAMS
----- RULE_TYPE
----- TYPE: int
------ IDENTIFIER: size
----- )
----- {
-- RULE_BODY
--- RULE_VARIABLE
----- RULE_TYPE
----- TYPE: float
--- IDENTIFIER: values
--- IDENTIFIER: []
--- =
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: new
---------- IDENTIFIER: float
---------- IDENTIFIER: []
Syntax Error 1400 at line 36, token: ;
Recovered: Missing array size specification
-- ;

--- RULE_FOR
--- for
--- (
--- RULE_VARIABLE
----- RULE_TYPE
----- TYPE: int
--- IDENTIFIER: i
--- =
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 0
-- ;
--- RULE_EXPRESSION
```

```
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ <
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: size
Syntax Error 41 at line 37, token: i
Recovered: Missing semicolon after condition in for loop
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
------ ++
--- )
---- RULE_STATEMENT_BLOCK
---- {
-- RULE_BODY
--- RULE_ASSIGNMENT
--- IDENTIFIER: values
--- IDENTIFIER: [
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
--- ]
--- =
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: i
-------- *
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- LITERAL: 2.5
-- ;
---- }
```

```
--- RULE_RETURN
--- return
--- RULE_EXPRESSION
-------- RULE_A
--------- RULE_B
---------- RULE_C
---------- IDENTIFIER: values
-- ;
----- }
---- }
Parsing completed with recovery.
Total errors found: 12
```