# 词法分析程序测试方案

## 测试目标

测试关键字、特殊符号、标识符和整数的正确识别，和报错。

## 自定义测试用例

### 关键字

测试用例 `custom-test-keyword.txt`：

```
cut.System.out.println.cut
System.out.println.cut
cut.System.out.println
System.out.printlnx
```

测试结果-Token序列：

```
IDENTIFIER, cut
DOT, .
PRINTLN, System.out.println
DOT, .
IDENTIFIER, cut
PRINTLN, System.out.println
DOT, .
IDENTIFIER, cut
IDENTIFIER, cut
DOT, .
PRINTLN, System.out.println
IDENTIFIER, System
DOT, .
IDENTIFIER, out
DOT, .
IDENTIFIER, printlnx
```

测试结果-报错信息：无

### 特殊符号

测试用例 `custom-test-punc.txt`：

```
#
+>
''
System.out.println
( 1 << 5 ) && 1
( 1 << 5 )&(1 < 4)
```

测试结果-Token序列:

```
ERROR, #
POSI, +
ERROR, >
COMMA, ,
COMMA, ,
PRINTLN, System.out.println
LT_PAREN, (
INTEGER_LITERAL, 1
LT, <
LT, <
INTEGER_LITERAL, 5
RT_PAREN, )
AND, &&
INTEGER_LITERAL, 1
LT_PAREN, (
INTEGER_LITERAL, 1
LT, <
LT, <
INTEGER_LITERAL, 5
RT_PAREN, )
ERROR, &(
LT_PAREN, (
INTEGER_LITERAL, 1
LT, <
INTEGER_LITERAL, 4
RT_PAREN, )
```

测试结果-报错信息:

```
(0:0): Error: the character `#` is not existed lexically.
#
^

(1:1): Error: the character `>` is not existed lexically.
+>
 ^

(5:11): Error: the AND operator is `&&`, not single `&` or `&(`.
( 1 << 5 )&(1 < 4)
          ^
```

# 标识符

测试用例 `custom-test-identifier.txt` :

```
identifier
public
publicx
underline_*
_APPosition_Input
1flow1
this_is_a_long_identifier_whose_length_is_very_long_designed_to_test_the_lexer_i
f_it_handles_this_situation_correctly
```

测试结果-Token序列：

```
IDENTIFIER, identifier
PUBLIC, public
IDENTIFIER, publicx
ERROR, underline_*
ASTER, *
ERROR, _
IDENTIFIER, APPosition_Input
INTEGER_LITERAL, 1
IDENTIFIER, flow1
IDENTIFIER,
this_is_a_long_identifier_whose_length_is_very_long_designed_to_test_the_lexer_i
f_it_handles_this_situation_correctly
```

测试结果-报错信息：

```
(3:10): Error: after the underline must be a letter or digit, not `*`.
underline_*
          ^

(4:0): Error: the underline character should not existed here.
_APPosition_Input
^
```

## 整数

测试用例 `custom-test-integer.txt`：

```
123
0123
3.14159265354
.01
0.0
if (2x < 4) x = x + 1
```

测试结果-Token序列：

```
INTEGER_LITERAL, 123
INTEGER_LITERAL, 0123
INTEGER_LITERAL, 3
DOT, .
INTEGER_LITERAL, 14159265354
DOT, .
INTEGER_LITERAL, 01
INTEGER_LITERAL, 0
DOT, .
INTEGER_LITERAL, 0
IF, if
LT_PAREN, (
INTEGER_LITERAL, 2
IDENTIFIER, x
LT, <
INTEGER_LITERAL, 4
RT_PAREN, )
```

```
IDENTIFIER, x
EQ, =
IDENTIFIER, x
POSI, +
INTEGER_LITERAL, 1
```

测试结果-报错信息： 无

## 默认测试用例

包括 `test1.txt` 至 `test7.txt` 的token序列和报错信息，分别在 `testcase` 中的 `.out` 和 `.err` 文件中。