编译原理词法分析器说明文档

**词法分析器功能**

读取java代码类型的文本，对其中的内容进行词法分析，识别保留字，数字，字符串等，并输出Token序列，包括内容和类型。

**分析器思路**

1. 针对要识别的词给出RE
2. 构造出每个RE对应的NFA
3. 根据NFA转换成DFA
4. 根据DFA进行代码实现

**词语类型**

**保留字**

类别为Reserved Word，包括void, class, public, private, protected, for, if, else,while, do, int, double, char, boolean, String, new, try, catch, static, return, this,main, switch, case, break, continue

**标识符**

类型为Identifier

**整数**

类型为Integer

**小数**

类型为Double

**操作符**

类型为Operator，包括+,++,+=,-,--,-=,\*,\*=,/,/=,|,||,&,&&,^,=,==,!,!=

**标点**

类型为Punctuation，包括 . , ; : () [] {}

**注释**

类型为Note，包含//型注释和/\*…\*/型注释

**字符串**

类型为String

**正则表达式**

Identifier->letter (letter|digit)\*

digit->0|1|2|3|4|5|6|7|8|9|0

letter->a|b|c|d|…|z

Integer->digit digit\*

Double->Integer . digit digit\*

Operator->+|++|=+|-|--|-=|\*|\*=| / |/=|=|==|&|&&| | | || |^| ! | !=

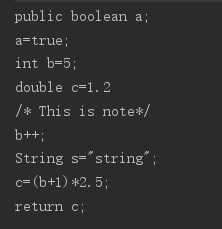
Punctuation->. | , | : | ; | ( | ) | [ | ] | { | }

ReservedWord-> void | class | public | private | protected | for | if | else |while | do | int | double | char | boolean | String | new | try | catch | static | return | this | main | switch | case | break | continue

Note->(// (letter|digit|punctuation|operator)\*) | (/\* (letter|digit|punctuation|operator)\* \*/)

String->” (letter|digit|punctuation|operator)\* “

输入输出示例

输入：

输出：