语言结构：顺序结构（赋值语句、输入、输出）、选择语句（if-else）、循环结构（while、for）。这些语句结构和C语言的结构一样，允许嵌套。

支持break，continue

表达式为关系表达式、算术表达式、逻辑表达式，运算的优先级为：算术运算、关系运算，并服从左结合规则。

算术表达式包括整数和实数上的运算、变量及“（）”、“”、“+”、“-”、“/”，运算符的优先级顺序为：“（）”大于“”和“/”大于“+”和“-”。

关系运算符包括：“<”、“>”、“==”、“<>”、“<=”、“>=”。

逻辑运算支持：”&&”、“||”

一条语句以“；”结束；程序由一条语句或者由“{”和“}”嵌套表达的复合语句。

支持多行注释（使用“/\*”和“\*/”）。

支持单行注释（使用“//”）

支持数组运算，数组的下标必须是正整数，使用“[”和“]”表示数组下标。

支持数据类型包括int，char, real

字符支持特殊转义字符：’\t’, ‘\r’, ‘\n’

变量的使用之前需要先声明，声明的方式和C语言一样。

十进制的整数与实数。

标识符（由数字、字母和下划线组成的串，但必须以字母开头、且不能以下划线结尾的串）。

|  |  |
| --- | --- |
| 单词符号 | 种别码 |
| if |  |
| else |  |
| while |  |
| for |  |
| break |  |
| continue |  |
| print |  |
| scan |  |
| + |  |
| - |  |
| \* |  |
| / |  |
| // |  |
| /\* |  |
| \*/ |  |
| \t |  |
| \n |  |
| \r |  |
| ( |  |
| ) |  |
| < |  |
| > |  |
| = |  |
| == |  |
| <> |  |
| <= |  |
| >= |  |
| ; |  |
| ‘ |  |
| “ |  |
| int |  |
| real |  |
| char |  |
| [ |  |
| ] |  |
| { |  |
| } |  |
| && |  |
| || |  |
| , |  |
| identifier |  |
| inumber |  |
| fnumber |  |
| def |  |
| return |  |

程序整体框架：

LA:

FileExecutor:

----readTestFile： 逐行读取文件

----replaceBlank： 处理换行、制表符

----readWord： 切割读取的每行，形成字符

----isOptsOrEnd: 判断是否为切割字符

JudgeTools:

----isIdentifier: 判断单词是否为合法变量名

----isKeyWord: 判断单词是否为关键字

----isFNumeric: 判断单词是否为合法浮点数字

----isINumeric: 判断单词是否为合法整型数字

LAnalyzer:

----execute: 进行词法分析

----show: 打印分析获得的单词信息

LATest:

----main: 进行词法分析测试

算法说明：

步骤一：替换文件中的\t,\r,\n为空格，文件逐行读取，对每行字符串执行步骤二。

步骤二：遍历行字符串中的字符，将字符分为普通字符和切割字符，通过切割字符将原字符串切割为单词，收入单词组，进入步骤三。

步骤三：对步骤二中获得的单词进行判断，是否为关键词，是否为数字或合法变量，若都不满足，标记该单词为非法字词，记录错误信息，否则，根据判断所得类别，标记单词并记录信息。其中对于注释关键词，进行标记，若为“//”，忽略同行单词，若为“/\*”，忽略单词直到遇到“\*/”

测试说明：