1. 概述
2. 词汇表 非常细致，单词符号，类别（关键字，运算符，编码，编码表（一符一码或一类一码））
3. 词法规则的形式化定义，正规式或自动机或巴克斯诺尔泛式
4. 选做（对应词法分析器的语义动作，自然语言说明（表格或者分条目说明））

Mini语言

肖健 吴晨

1. Mini语言概述
2. 不允许重载
3. 只有类，没有接口，无继承，不能定义内部类，必须有初始化函数
4. 四种变量：整形(int)、布尔型(bool)、浮点型(double)、字符串(string)
5. 语句：赋值语句(=)，打印语句(print)、判断语句(if，else)、循环语句(while), 代码块({、})，括号（）
6. 运算符：+、-、\*、/、<、>、||、!(表示非)，%(取模)、<=、>=，==，!=
7. 标识符：字母开头，由字母、数字和下划线组成
8. 字符串：以“”开头和结尾
9. 注释：只支持行注释，以//开头
10. 语句必须以；结束
11. 暂不支持异常处理
12. 入口函数为main函数
13. 支持跨文件的类调用，使用import进行导入
14. 保留字与终结符
15. 类型标识符：bool, int, double, string, void, false, true
16. 类标识符：class，new(创建对象)
17. 内置函数：print
18. 控制语句：while，return, if, else
19. 运算符：+、-、\*、/、&&、||、%、！、.(点号)、(逗号，)、>、<、=
20. 导入符号：import

三、符号表

|  |  |  |  |
| --- | --- | --- | --- |
| 保留字 | 识别编码 | 类别 |  |
| bool | 1 | Type Indicator |  |
| int | 2 | Type Indicator |  |
| double | 3 | Type Indicator |  |
| string | 4 | Type Indicator |  |
| void | 5 | Type Indicator |  |
| false | 6 | Bool value |  |
| true | 7 | Bool value |  |
| class | 8 | Class Indicator |  |
| new | 9 | Allocator |  |
| print | 10 | Built-in function |  |
| return | 11 | Return |  |
| if | 12 | Control |  |
| else | 13 | Control |  |
| break | 14 | Control |  |
| + | 15 | Arithmetic Operator |  |
| - | 16 | Arithmetic Operator |  |
| \* | 17 | Arithmetic Operator |  |
| / | 18 | Arithmetic Operator |  |
| % | 19 | Arithmetic Operator |  |
| || | 20 | Logic Operator |  |
| && | 21 | Logic Operator |  |
| ! | 22 | Logic Operator |  |
| < | 23 | Logic Operator |  |
| > | 24 | Logic Operator |  |
| = | 25 | Assignment Operator |  |
| ( | 26 | Bracket |  |
| ) | 27 | Bracket |  |
| { | 28 | Block Indicator |  |
| } | 29 | Block Indicator |  |
| ; | 30 | Semicolon |  |
| , | 31 | Comma |  |
| “ | 32 | Quotation |  |
| 数字 | 33 | Number |  |
| 标识符 | 34 | Identifier |  |
| 字符串 | 35 | STRING |  |
| import | 36 | IMPORT |  |

四、词法规则的描述

**说明**：我们打算在词法分析阶段不区分保留字，等识别完分词后再判断是否为保留字

开始符号为s

s**->**number **|** identifier **|** string

number**->**(-)?[1-9][0-9]\* **|** (-)?0.[0-9]+ **|** (-)[1-9][0-9]\*.[0-9]+ **|** 0

identifier **->**[a-zA-Z\_]+[a-zA-Z0-9\_]\*

string **->** “[a-zA-Z0-9\_ ]+” //允许有空格