**编译系统课程实验报告**

**实验1：词法分析**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 刘璟烁 | | 院系 | | 计算机科学与技术 | | | 学号 | | |  | | |
| 任课教师 | |  | | | | 指导教师 |  | | | | | | |
| 实验地点 | |  | | | | 实验时间 | 2021-4-10 | | | | | | |
| 实验课表现 | | 出勤、表现得分 | |  | | 实验报告  得分 |  | | 实验总分 | | |  | |
| 操作结果得分 | |  | |
| **一、需求分析** | | | | | | | | | | 得分 | | |  |
| 要求：阐述词法分析系统所要完成的功能  词法分析任务是编译的第一个阶段，也是编译器最基础的模块，它由词法分析器完成。词法分析器读入表示源程序的字符流，按照程序功能等价的要求，从左到右对这个字符流进行扫描，以正则文法、有穷自动机为基本原理**切分出其中由字符组成的单词并生成对应的单词序列，剔除其中的空格、注解等不影响程序语义的字符；**  另外词法分析系统还需要**检测并报告、定位源程序中存在的词法错误**。 | | | | | | | | | | | | | |
| **二、文法设计** | | | | | | | | | | 得分 | | |  |
| 要求：对如下内容展开描述   1. 给出各类单词的词法规则描述（正则文法或正则表达式）：   我们这里选择使用正则表达式来表述对各类单词的词法规则。   1. 标识符：   Identifier = (char | \_)(char | digit | \_)\* (注：char指代英文字符，digit指代单个0-9范围内的数字)   1. 无符号整数：   Decimal = (1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9)(0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9)\* | 0  Oct\_digit = 0(1 | 2 | 3 | 4 | 5 | 6 | 7) (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7)\*  Hex\_digit = 0x(1 | ... | 9 | a |...| f | A | ... | F)(0 | 1 | ... | 9 | a |...| f | A | ... | F)\*   1. 浮点数（科学计数法）   Float = Decimal (.(Decimal) | )(E(+ | - | )Decimal | )   1. 字符串常数：   String = (‘|”)(char)\*(‘|”)   1. 注释：   Annotation = /\* (any character)\* \*/ (注：any character 指代任意符号)   1. 关键字、运算符、界符(这三类单词都是有限的，可通过枚举表示)：   Keywords = if | else | ......  Operator = + | - | ......  Separator = { | } | ......   1. 各类单词的转换图   A.无符号整数常量：  截屏2021-04-06 下午10.29.52  B.浮点数  截屏2021-04-06 下午10.09.09  C.注释  截屏2021-04-06 下午9.58.56  D.标识符  截屏2021-04-06 下午10.12.40  E.字符串常量  截屏2021-04-06 下午10.37.48 | | | | | | | | | | | | | |
| **三、系统设计** | | | | | | | | | | 得分 | | |  |
| 要求：分为系统概要设计和系统详细设计。   1. 系统概要设计：给出必要的系统宏观层面设计图，如系统框架图、数据流图、功能模块图等以及相应的文字说明。   系统通过调用工具类Lexer实现词法分析功能，我们将在(2)中给出该类的数据结构和主要功能函数，并仔细分析主要功能Lexer.scanner的程序结构，目前给出系统宏观设计图如下：  截屏2021-04-25 上午10.08.18   1. 系统详细设计：对如下工作进行展开描述   首先给出Lexer类的UML图描述核心数据结构的设计：  截屏2021-04-25 上午10.24.24   * 核心数据结构的设计 * 对于一段输入的数据流，我们在逻辑上设定一个指向下一个将被扫描字符的读入指针，并用变量row\_num和col\_num分别表示该读入指针的行位置和列位置； * octal\_set和hexad\_set分别为程序判断token是否为8进制和16进制时所需要用到的集合，通过直接给出这两个变量可以优化程序执行时间，防止在每次判断时都要生成一遍； * KEYWORD、OPERATOR、SEPARATOR分别为C语言关键字、操作符、分隔符的集合； * spcie\_dic字典用于映射上述关键字、操作符、分隔符到它们的种别码； * signal\_dic字典用于映射新加入的id到它的属性值； * 主要功能函数说明 * getchar(buffer)方法读入从数据流buffer中读入下一个字符并更新类属性row\_num、col\_num的值，直到buffer结尾输出EOF； * retract(buffer)方法更改指针位置使指针会退一个字符； * error\_handle(message,line\_num)方法根据参数提示信息message和错误发生行line\_num在控制台输出错误； * scanner(buffer)为程序核心部分，循环读取buffer中的内容直到EOF，其具体结构将会程序核心部分流程图中给出; * client(c\_file,output\_file)方法从待编译文件c\_file中读入数据流，并调用scanner方法切分出词法单元，按照给定格式写入到文件output\_file中； * 程序核心部分的程序流程图   未命名文件(12) | | | | | | | | | | | | | |
| **四、系统实现及结果分析** | | | | | | | | | | 得分 | | |  |
| 要求：对如下内容展开描述。   1. 系统实现过程中遇到的问题； 2. 关于如何逐字符读取输入的数据流的问题：经过分析，词法分析器在检测出词法错误时需要报告错误发生的行号。因此为了便于实现，我们将输入文件按行切分，并记录读入指针的行位置和列位置逐个读取字符。 3. 关于词法单元、注释、错误提示的输出：我们选择将词法单元串输出到指定文件，并在控制台输出错误提示以及检测到的注释。 4. 关于数字的识别：在程序设计之初尝试分块实现浮点数和整形常量的识别，但由于无法预测第一个数字后面是否有e或.的存在，因此需要将两者的DFA图进行合并，重新设计程序结构。 5. 针对某测试程序输出其词法分析结果；   在本次试验中实现了测试程序lexical\_input.c，并作为程序输入，其词法分析结果已经写入out\_put.txt中，先展示其部分结果：  截屏2021-04-25 上午11.23.43  其中关键字、分隔符、操作符的种别码以大写英文字母串的形式给出，但在程序内部以整形存储。   1. 输出针对此测试程序对应的词法错误报告；   在标准输出中可显示测试程序的词法错误报告如图：  截屏2021-04-25 上午11.30.20   1. 对实验结果进行分析。 2. 通过观察标准输出以及输出的词法单元序列可知，本次试验中实现的词法分析器不仅可有效识别给定的分界符、操作符以及32个C常用关键字，还能够有效识别并分辨出数字的类型，分别以FLOAT、DECIMAL、OCTAL、HEXADECIMAL表示浮点数、十进制整数、八进制整数、十六进制整数，同时能够有效识别出标识符，并对第一次定义的标识符以特定属性值存储到类属性signal\_dic中； 3. 同时，实验结果表示该程序可以有效识别出预先设定的五种词法错误，分别包括非法字符、非法浮点数、非法十六进制数、字符串常量未封闭、注释未封闭等五种错误； 4. 程序满足自动跳过空行、空格、制表符等功能；   注：其中的测试样例自行产生。 | | | | | | | | | | | | | |
| 指导教师评语：  日期： | | | | | | | | | | | | | |