编译原理词法分析实验报告

选课序号： 19 专业班级： 计算机科学与技术2019-3班

姓名：\_胡聿鑫\_学号：\_ 2220192813\_\_\_\_\_\_日期： 2021年11月2日

# 实验目的

设计编写一个词法分析程序，加深对词法分析原理的理解，熟练掌握词法分析程序的实现方法，要求能够识别TEST语言的单词并完成错误处理。

# 实验内容及要求

(一) 词法分析程序

输入字符流文件，输出单词流文件。根据DFA采用直接编程法或者表驱动方法实现词法分析程序。

1. 保留字对大小写不敏感，保留字表的查找采用折半查找方法；
2. 注释格式为：/\*……\*/，注释不能嵌套；
3. 删除注释、空格等无用符号；
4. 能发现词法错误，并将错误信息输出到屏幕上，自定义错误处理模式；
5. 如果采用直接编程法实现词法分析程序，可以将某些类型的单词识别过程利用表驱动算法实现。

# 实验相关理论知识

词法分析程序的主要目标是从字符串表示的源程序中识别出具有独立意义的单词符号，其基本思想是根据扫描到的单词符号的首个字符种类，输出相应的单词符号。

符号种类包括保留字、标识符、数字、单分界符和双分界符。其中保留字和分界符均采用一字一类编码，标识符和常量采用各自用一个类别码。单词流的输出采用二元组的形式表示，即（单词类别，单词值）。若想确定字符是字母、数字还是TEST语言无法识别的符号，则可以通过ASCII码进行判断。

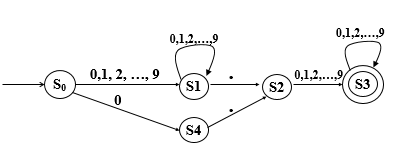
一旦一个词法错误被发现，词法分析器不会停止，会采取措施来继续词法分析的过程，开始读取下一个字符。错误的类型包括未知字符、以数字开头的标识符、含有多个小数点的数字和缺失的注释符号等。

程序设计语言应当使用正则文法，其描述的语言使用DFA表示。通过使用状态图，可以更加直观地反应词法分析程序的设计思路。假设A是一个DFA，a1 a2 a3…an是一个字符串，如果存在一个状态序列(S0,S1,S2,…,Sn)，它满足：



其中S0是开始符号，Sn是接收状态之一，若字符串a1 a2 a3…an可被DFA A 所接受，则表示为L(A)。

针对DFA定义的字符串集：所有的字符串集中的字符串都是被DFA A所接受的被称为A定义的字符串集，以无符号实数的DFA为例：



# 实验过程、运行结果

首先定义保留字表{"if","else","for","while","do","int","read","write","sum"}，单分界符"+-\*(){};,:"，双分界符"><=!"。单词的内部表示存储在词法单元token中。然后在词法分析函数中给出TEST源程序文件路径和要输出的目标文件的位置，接着逐个字符判断直至全部读取完毕。

在词法分析函数中，先要判断是否遇到空格、制表符或换行符，若遇到则需要将其跳过，读取下一字符。使用isalpha函数判断当前字符是否为字母，若是则用isalnum函数判断是否为字母数字的组合，若是则组合标识符，否则结束组合并将结果保存到token中，然后读取下一字符。标识符识别结束后，先将token中的字符均转换成小写，再用折半查找识别保留字，将找到的保留字符号进行输出，查找结束后剩余符号即被作为标识符输出。

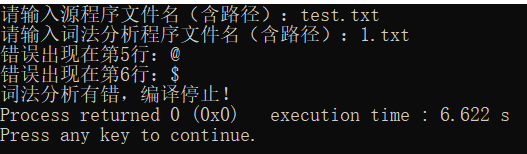
字母判断结束后，则进入数字的判断使用isdigit函数对数字组合，组合后的整数保存在token中，最后输出整数符号。然后针对单分界符的处理，则需要将字符和数组singleword（存放了所有的单分界符）中的元素进行比较，若比较成功则将此单分界符输出。双分界符和单分界符的处理方法类似，只不过比较的数组变成了doubleword。

最后完成删除注释和错误处理的相关操作后，即可进入主函数的编写，调用词法分析函数最终完成输出。

该程序流程图如下：

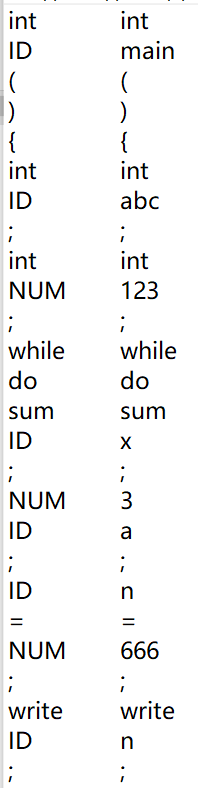
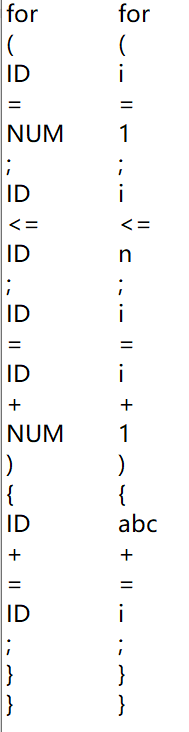


最终屏幕显示输出的结果如下图所示：



需要注意的是，尽管显示词法分析错误，但词法分析程序文件中仍然会出现整个程序的词法分析结果，对于出错的部分会自动跳过。

其中词法分析程序文件的内容如下：

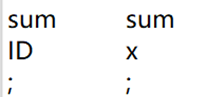
# 实验总结

实验初期对于程序的编写无从下手，因此重新复习了整个词法分析的相关知识，并查阅了有关资料，最终完成了用C语言实现词法分析器的工作。现将实验过程中的心得体会及遇到的问题总结如下：

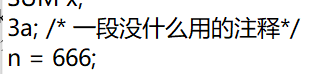
为实现保留字对大小写不敏感这一功能，需要词法分析函数中在判断保留字之前，将字符串强制转换成小写，这样才能保证无论保留字是全部大写、全部小写还是大小写混合，均能够被识别为保留字。具体做法为：使用C语言自带的转换函数strlwr，将token中的字母均转换成小写只需要strlwr(token)即可实现。例如在TEST源程序中定义如下语句（sum为拓展的关键字）：



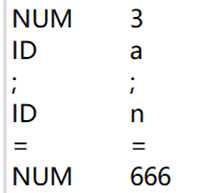
经词法分析程序识别后，SUM仍被识别为保留字sum，其结果如下图所示：



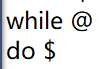
要跳过注释，则需要逐个字符判断，首先在连续遇到“/”、“\*”这两字符时，开始删除注释，直至再次遇到字符“\*”、“/”时停止处理注释。TEST程序中任意编写一段注释：



最后在输出文件中，不会显示任何与注释有关的信息，而是直接跳过注释，继续向下进行词法分析，如下图所示：



若想显示错误所在行，可以先定义一个全局变量，每换行一次则使该全局变量+1，最后在识别所有类别字均失败时，同时输出行数及该错误字符。假设在TEST源程序中的5，6行各加入一个程序无法识别的符号，如下图所示：



词法分析时会显示该错误的所在，同时显示错误的内容，如下图所示：



折半查找的具体实现方式是：使用strcmp函数进行条件判断，利用char数组读入字符，并在字符串末位尾添加‘\0’作为结束符。在进行折半查找的过程中要匹配关键字，则需要将已排序的关键字（初始化阶段完成）进行首尾比较，匹配成功则说明该字符是保留字，将其输出，否则查找范围减半后继续进行查找。

本次实验使我对词法分析程序的整体设计思路和实现方法有了更深的理解，在完成基本功能的基础上，实现了对关键字的折半查找，和关键字的大小写不敏感处理，同时也能够显示出错误所在行数。但不足之处在于：没有完成用表驱动算法实现单词的识别，错误处理部分也无法对错误进行自动修正。对此我会加强关于表驱动算法的学习，不断尝试用表驱动方法完成词法分析程序。错误的自动修正也将成为我编译原理课程学习的长期目标，争取可以早日实现该功能。