实验一 TINY语言的词法分析

实验目的

构造tiny语言的词法分析器（扫描器），要求利用第三方的lex工具进行构造。实验结果：构造出的扫描器，能够读入教材样例中给出的tiny语言的示例代码，分解成token输出。

Experimental Requirement

Build the lexical parser (a scanner) of TINY language, using the Lex tool of third party. The experimental result Scanner can receive the sample program of Tiny language, and output is a sequence of tokes that are defined by regular expression.

预习要求：

Preview requirements:

1.查阅各种专业文献，针对任意二种现代程序设计语言编译器的词法分析技术，找出各自的标志性的特征点，并加以比较分析和比较，指出其优缺点，并试图给出自已的结论。

Referring to professional literatures, aiming at the lexical analysis technology of any two modern programming language compilers, finds out their respective features, compares them, points out their advantages and disadvantages, and tries to give your Evaluation and conclusions.

词语分析技术是编译原理的关键一个步骤也是不可缺一的步骤，它将输入的程序逐个成字符扫描转换成标记，这里举例Python和Java的词法分析各自的优点和缺点。Python是一个高级语言简单易懂也是我们初学者最容易上手的，Python的编译器使用正则表达式的词法分析，首先先逐个字符读进输入程序然后用正则表达式标记。Python还有一个主要的有点就是灵活性，允许自定义标记。Java的词法分析主要的一个就是其精度，可以处理复杂的语法，并能识别正则表达式更复杂的标记，与其Python对比Java会更难维护以及修改。我个人认为Python和Java都有各自的优点，我们要灵活运用对应的程序代码才是最主要的，Python的代码主要是灵活性和容易修改，而Java的提供精度和控制。主要要看我们的项目的要求来决定使用Python还是Java。

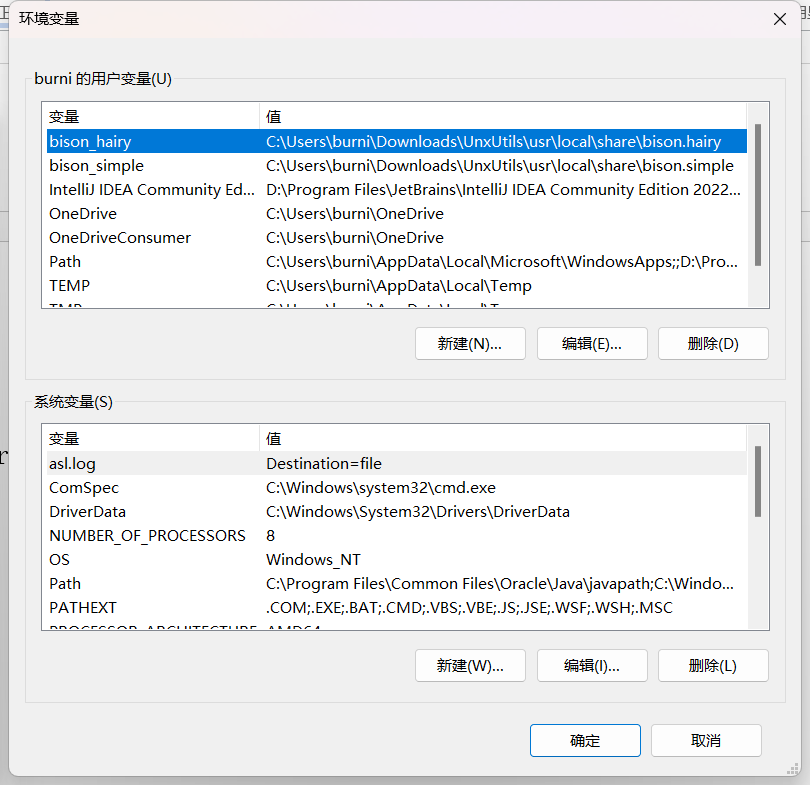
2.请描述自已拟定的实验计划和步骤，以及对输入和输出的设计。

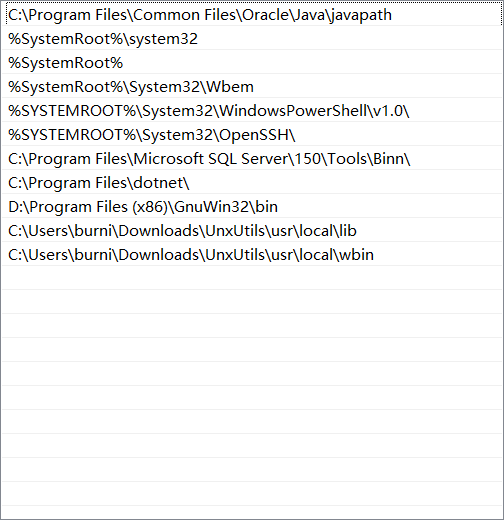
Please design your experimental plan and procedures, as well as the design of input and output of the SCANNER

TINY的程序结构很简单，它在语法上与A d a或P a s c a l的语法相似：仅是一个由分号分隔开的语句序列。另外，它既无过程也无声明。所有的变量都是整型变量，通过对其赋值可较轻易地声明变量（类似F O RT R A N或B A S I C）。它只有两个控制语句： i f语句和r e p e a t语句，这两个控制语句本身也可包含语句序列。I f语句有一个可选的e l s e部分且必须由关键字e n d结束。除此之外，r e a d语句和w r i t e语句完成输入/输出。在花括号中可以有注释，但注释不能嵌套。TINY的表达式也局限于布尔表达式和整型算术表达式。布尔表达式由对两个算术表达式的比较组成，该比较使用<与=比较算符。算术表达式可以包括整型常数、变量、参数以及4个整型算符+、－、\*、/，此外还有一般的数学属性。布尔表达式可能只作为测试出现在控制语句中——而没有布尔型变量、赋值或I / O。

3.选择三方工具，阅读其使用说明，并尝试建立起工作环境。

Choose the LEX tools, read their instructions, and try to establish a working environment.

找到编辑系统环境->环境变量-> 

变量里要添加bison.hairy和bison.simple的位置，然后系统变量要添加

创建文本然后改名lex.l打开cmd。如果没报错就配置成功了