郑州轻工业大学本科

**编译原理课程设计总结报告**

**设计题目**： FOR语句的四元式结构及其语法制导

**学生姓名**： 程军伟 原彬贺

**系 别**： 计算机与通信工程学院

**专 业**： 计算机科学与技术

**班 级**： 计科20-02

**学 号**：

**指导教师**： 马吉明 韩丽

**成 绩：**

**时 间：**2023**年**6**月**12**日至**2023**年**6**月**18**日**

说明

1、课程设计期间，学生应按教学计划，将每天的学习情况（包括学习的内容、遇到的问题及解决办法 、心得体会等）如实进行记录。

1. 结束时，根据课设内容和学习记录写报告。

3、指导老师应综合考虑学生的学习态度、报告内容和实际操作情况等，给出评语和成绩。

课程设计撰写格式

1、课程设计总结报告一律采用A4的纸张，打印正文用宋体小四号字；版面页边距：上2.5 cm,下2.5cm,左3cm,右2cm；页码用小五号字底端居中；左边装订。

2、课程设计报告一般由以下几部分组成：A.任务书；B.目录；C.正文；D.参考文献；E.附录。

3、报告正文撰写的题序层次格式；

|  |  |
| --- | --- |
| 第一种 | 第二种 |
| 第一章（居中、小二号黑体字） | 1.（居中、小二号黑体字） |
| 第一节（顶格、小三号黑体字） | 1.1（顶格、小三号黑体字） |
| 一、（顶格、四号黑体） | 1.1.1（顶格、四号黑体） |
| （一）（顶格、小四号黑体） |  |

郑州轻工业大学

**课程设计任务书**

**题目： FOR语句的四元式结构及其语法制导**

**专业班级：** 计算机科学与技术20-02

**学号：**

**姓名：**

**主要内容、基本要求、主要参考资料等：**

**主要内容**：

以语句的四元式中间代码及其语法制导生成为例分析语法制导生成方法。给出一种循环语句的四元式结构以及语法制导生成过程，包括词法分析、语法分析以及中间代码生成三个阶段。

**基本要求**：

构造出程序设计语言各语法单位的SLR(1)分析表，能够对输入的源程序做出正确的语法分析并输出清晰的结果；给出设计说明和程序结构框架，阐明设计思路、原理和方法；设计和调试过程要规范化；运行系统并要通过验收，讲解运行结果，并回答指导教师的提问；提交课程设计报告。

**主要参考资料**：

1. 张素琴、吕映芝、蒋维杜、戴桂兰编著.《编译原理》（第2版）[M].清华大学出版社，2012年；
2. 冯博琴等译.《编译原理及实践》[M].机械工业出版社，2013年；
3. 王生原等编著.《编译原理》（第3版）[M].清华大学出版社，2015年；

**完 成 期 限：** 一周

**指导教师签名：**

**专业负责人签名：**

**2023年 6月 18日**

设计分工

组长学号及姓名：

分工：共同规定符号表，搭建框架，

LR分析器设计、语义分析及四元式生成，课设报告，PPT制作，答辩。

组员1学号及姓名：

分工：共同规定符号表，搭建框架，

符号表整理，词法分析，PPT制作，课设报告，答辩。

组员2学号及姓名：

分工：共同规定符号表，搭建框架，

语法分析，绘制流程图，计算LR分析表，课设报告，答辩。

**目录**

[1. 课程设计目的 2](#_Toc25608)

[2. 课程设计内容 2](#_Toc15987)

[3. 课程设计原理 3](#_Toc32767)

[4. 系统需求分析 7](#_Toc871)

[4.1 “嘻咖咖”编译程序总体介绍 7](#_Toc12728)

[4.2 功能需求 8](#_Toc2838)

[4.3 用到的主要关键词： 9](#_Toc26683)

[4.4 汇编成可运行文件 10](#_Toc4732)

[4.5 “嘻咖咖”语言的文法 11](#_Toc5726)

[4.5.1 “嘻咖咖”语言分析表 11](#_Toc27575)

[5. 系统设计与实现 14](#_Toc19934)

[5.1 BY13编译程序主要功能模块 14](#_Toc12613)

[5.2 词法分析子程序 15](#_Toc6189)

[5.3 语法语义分析子程序 17](#_Toc25616)

[5.3.1 LR分析表 17](#_Toc8900)

[5.3.2 语法语义分析子程序 20](#_Toc21246)

[5.3.3 汇编代码生成 22](#_Toc17151)

[6. 系统测试与运行结果分析 22](#_Toc24543)

[6.1 简单编译测试 22](#_Toc13765)

[7. 心得体会 26](#_Toc5359)

[8.课程设计日志 28](#_Toc24975)

# 课程设计目的

（1）根据所掌握的编译原理课程的基本知识，编写出一个基于for循环的解释器器；

（2）增强阅读和编写程序的能力；

（3）理解词法分析、语法分析和语义分析在编译程序中的作用；

（4）掌握词法分析、语法分析和语义分析程序的实现方法；

（5）实现基于for循环的翻译器的编辑、编译、连接和运行。

# 课程设计内容

以语句的四元式中间代码及其语法制导生成为例分析语法制导生成方法。给出一种循环语句的四元式结构以及语法制导生成过程，包括词法分析、语法分析以及中间代码生成三个阶段。

该程序的单词符号定义（即支持的语句和运算）为以下几种：

1、标识符(ID)=letter ( letter | digit )\*

2、常数：整数型(NUM)=digit digit\*

3、浮点型(Float)=NUM . NUM

4、运算符(OP)：

5、数学运算：+、-、\*、/、%

6、关系运算：<、>、<=、>=、==、=

7、逻辑运算：&&，||，!

8、界符：(、)、[、]、{、}、;、，

9、数据类型(DT)：int，float，char，void

10、语句：赋值(=)，if，while，for

赋值语句(Assign)：

fu<ID> = <Expression>;

条件语句：

If<Expression>{

statement.

}(else{

statement.

})+

+：0或1

while<Expression>{

statement.

}

break;

continue;

11、矩阵(Matrix)

mat(ID)=|0000|

|0000|

|0000|

12、支持函数的定义、调用

13、支持复合语句，即 ｛｝ 包含的语句

14、注释： C 类型的多行注释 /\* \*/ 和 C++ 类型的单行注释 //

# 课程设计原理

3．1 翻译过程

词法分析是计算机科学中将字符序列转换为单词（Token）序列的过程。进行 语法分析的程序或者函数叫作词法分析器（Lexical analyzer,简称Lexer）,也 叫扫描器（Scanner）。词法分析器一般以函数的形式存在，供语法分析器调用。

词法分析是编译过程中的第一个阶段，在语法分析前进行°也可以和语法分析结 合在一起作为一遍，由语法分析程序调用词法分析程序来获得当前单词供语法分析使 用。简化设计、改进编译效率、增加编译系统的可移植性。词法分析是编制一个读单

词的过程，从输入的源程序中，识别岀各个具有独立意义的单词，即关键字、标识符、 常量、运算符、界限符五大类。并依次输出各个单词的类别及单词符'，单词的分类主要分为五类：

1. 关键字：由程序语言定义的具有固定意义的标识符。也称为保留字或基本字。

2. 标识符：用来表示程序中各种名字的字符串。

3. 常量：常量的类型一般有整型、实型、布尔型、文字型。

4. 运算符：如+、一、\*、/、++、=＞、＜=、一等。

5. 界限符：如逗号、分号、括号等。

词法分析器输出的单词符号常表示成如下的二元式：（单词种别，单词符号的属 性值）

3．2语法分析

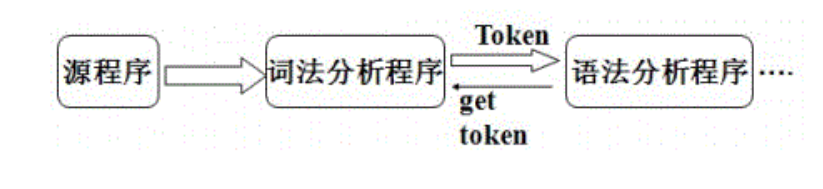
语法分析是编译过程的一个逻辑阶段。语法分析的任务是在的基础上将单词 序列组合成各类语法短语，如“程序”，“语句”，“表达式”等等.语法分析程序判 断源程序在结构上是否正确O源程序的结构由上下文无关文法描述。语法分析程 序可以用YACC等工具自动生成。

语法分析是编译程序的核心部分，其主要任务是确定语法结构，检查语法错误，报 告错误的性质和位置，并进行适当的纠排工作。语法分析的主要工作：是识别由词法 分析给出的单词序列是否是给定的正确句子（程序）。语法分析常用的方法：自顶向下 的语法分析和自底向上的语法分析两大类。此次设计中语法分析中主要通过递归下降 分析法对语法分析处理过程进行控制，使輸出的四元式表示的翻译的工作有条不紊的 进行，同时识别语法分析中的语法错误。

递归下降法主要釆用自顶向下方法，即从文法的开始符号开始进行分析，逐渐推 导的往下构造语法树，使其树叶正好构造出所给定的源程序串。白顶向下方法的关键 是确定在推导过程中选择候选式的问题。当进行推导时，一个非终结符可能对应多个 产生式，这样我们就无法事先知道应该用哪个产生式，因此实用都作了一些限制。以 便在任何情况下都能确定应该用的产生式。

白顶向下的主要思想是从开始符岀发导出句型并一个符号一个符号地与给定终结 符串进行匹配。如果全部匹配成功，则表示开始符号可推导出给定的终结符串。因此 判定给定终结符号串是正确句子。

词法分析程序和语法分析程序的关系：



3.3 中间代码生成

中间代码，也称中间语言，是复杂性介于源程序语言和机器语言的一种表示形式。 为了使编译程序有较高的目标程序质量，或要求从编译程序逻辑结构上把与机器无关 和与机器有关的工作明显的分开来时，许多编译程序都采用了某种夏杂性介于源程序 语言和机器语言之间的中间语言。中间代码（语言）是一种特殊结构的语言，编译程 序所使用的中间代码有多种形式。按其结构分常见的有逆波兰式（后缀式）、三地址代 码（三元式、四元式）和树形表示（抽象语法树）、DAG表示。本次课程设计要实现 的是输出四元式表示。

3.4属性文法

对于文法的每个产生式都配备了一组属性的计算规则，称为语义规则。所谓语法 制导的翻译指的是在语法分析过程中，完成这些语义规则描述的动作，从而实现语义 处理。一个属性文法包含一个上下文无关文法和一系列语义规则，这些语义规则附在 文法的每个产生式上。形式上讲，属性文法是一个三元组：A=（G, V, F）,其中：

G:是一个上下文无关文法：

V:有穷的属性集，每个属性与文法的一个终结符或非终结符相连，这些属性代表与 文法符号相关信息：

F:关于属性的属性断言或一组属性的计算规则（称为语义规则）。断言或语义规则 与一个产生式相联,只引用该产生式左端或右端的终结符或非终结符相联的属性。

# 系统分析

编译过程采用LR(1)分析方法进行分析，当栈顶符号形成句柄的时候，且文法中又当前规则的时候，进行规约并对符号栈和状态栈做出相应的改变。当语法分析需要读取输入符号串时就调用词法分析程序；而当语法分析正确后，需要生成相应的目标代码。对源程序分别进行 词法分析、语法分析、语义分析、中间代码生成、目标代码生成和可执行程序生成，此外，用表格管理程序建立变量、常量和过程标识符的说明与引用之间的信息联系.用出错处理程序对词法和语法分析研究遇到的错误给出在源程序中出错的位置和错误性质，它的工作过程如下所示：

*输入源代码 -> 词法分析 -> 语法分析 -> 中间代码生成->汇编代码生成*

## 整体流程图

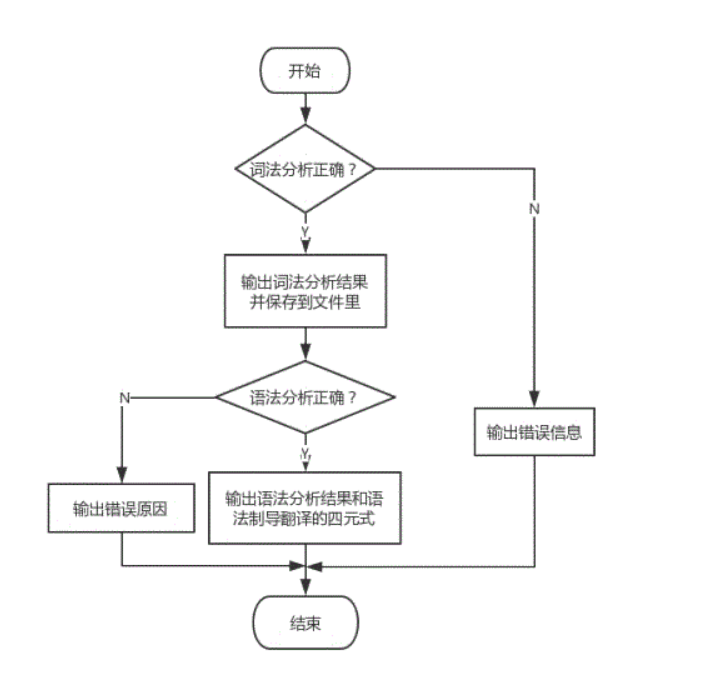
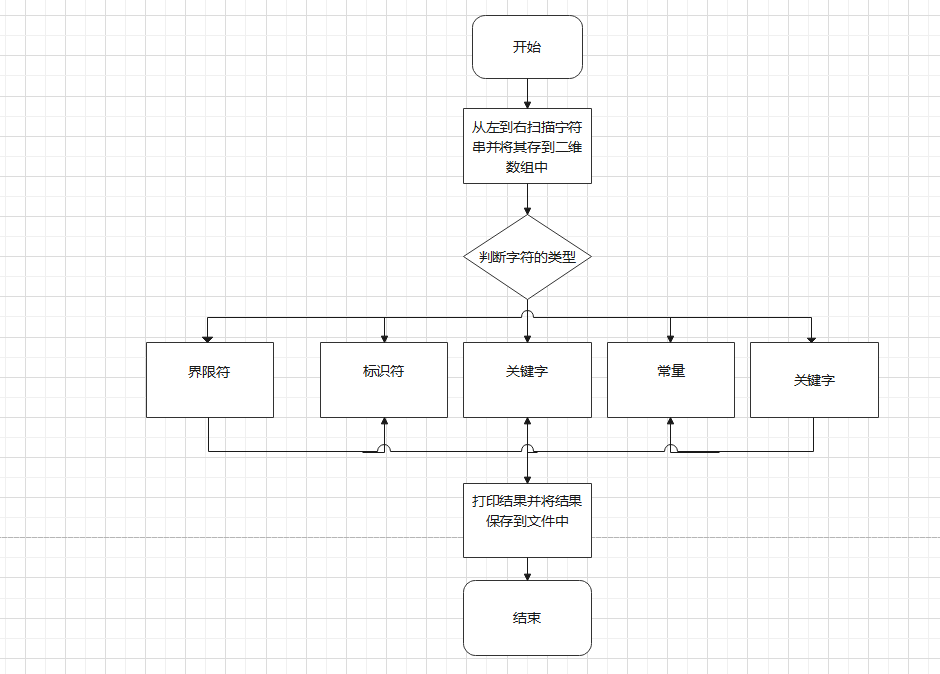
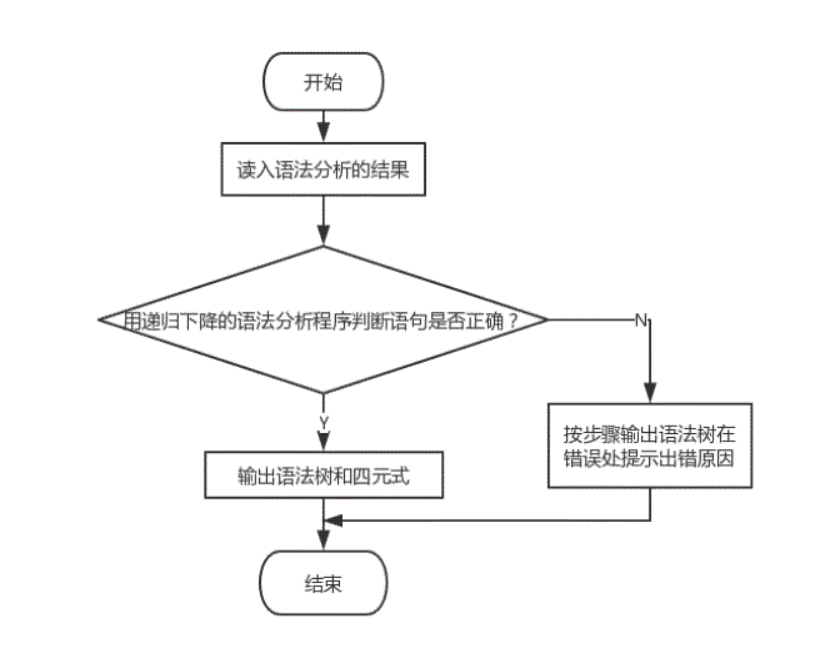


图 4‑1 编译程序总体介绍

## 词法分析流程图



## 语法制导翻译流程图



## 系统功能

（1）用 C++ 语言实现了类C语言，叫做“嘻咖咖”语言；

（2）采用LR(1)分析方法进行分析，当栈顶符号形成句柄的时候，且文法中又当前规则的时候，进行规约并对符号栈和状态栈做出相应的改变。当语法分析需要读取输入符号串时就调用词法分析程序。

（3）在声明中实现了对静态常量、变量和函数声明的支持；赋值语句的 = ，基本的数学运算 +、-、\*、/ ；

（4）使用函数实现了对过程的调用；

（5）使用 { } 实现复合语句；

（6）在循环分支语句中实现了if语句， if……else……语句， for(…,……,……)…语句，while……语句；

（7）关系表达式包括 ==(等于)、!=(不等于)、<、<=、>、>=。

（8）解析矩阵表达式

## 对LL(1)分析的描述

LL（1）：第1个L表明自顶向下分析是从左向右扫描输入串，第2个L表明分析过程中将用最左推导，1表明只需向右看一个符号便可决定如何推导即选择哪个产生式（规则）进行推导。从文法的开始符号出发企图推导出与输入的单词串完全相匹配的句子，若输入串是文法给定的句子，则必能推出，反之必然出错。

LL(1)优点：实现方法简单、直观，便于手工构造或自动生成语法分析器 。文法很 容易写出。

LL(1)缺点：对文法有一定得限制，要求推导过程中紧跟在飞终结符右边可能出现的终结符集不相交。另外在做语法制导翻译时中间代码的输出方案相对于LR法比较复杂。LR分析法的句柄即体现了算符的优先级，出现句柄即做相应的归约动作，中间代码很容易写出，实现很简单。LL(1)是做自顶向下推导，因此设计LL（1）的语法制导翻译输出中间代码很需要技巧，涉及到了判断符号的优先级。

## 语法分析表设计

文法(消除了左递归）及相应的选择符集

|  |  |  |
| --- | --- | --- |
| NO. | 文法 | SELECT集 |
| 0 | A->f(B){Y} | { f } |
| 1 | B->CDG | { ;,i } |
| 2 | Y->ε | { } } |
| 3 | Y->m=E | { m } |
| 4 | C->; | { ; } |
| 5 | C->i=E | { i } |
| 6 | D->iF; | { i } |
| 7 | F-><E; | { < } |
| 8 | F->>E | { > } |
| 9 | G->ε | { ) } |
| 10 | G->i=E | { i } |
| 11 | E->LM | { (,i } |
| 12 | M->+LM | { +} |
| 13 | M->-LM | { - } |
| 14 | M->ε | { ),; } |
| 15 | L->NP | { (,i } |
| 16 | P->\*NP | { \* } |
| 17 | P->/NP | { / } |
| 18 | P->ε | { +,-,),;} |
| 19 | N->i | { i } |
| 20 | N->(E) | { ( } |

## 预测分析表设计

表达式文法的预测分析表

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | f | i | m | ; | < | > | + | - | \* | / | ( | ) | { | } | = |
| A | 0 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| B | -1 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| C | -1 | 5 | -1 | 4 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| D | -1 | 6 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| E | -1 | 11 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 11 | -1 | -1 | -1 | -1 |
| F | -1 | -1 | -1 | -1 | 7 | 8 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| G | -1 | 10 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 9 | -1 | -1 | -1 |
| L | -1 | 15 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 15 | -1 | -1 | -1 | -1 |
| M | -1 | -1 | -1 | 14 | -1 | -1 | 12 | 13 | -1 | -1 | -1 | 14 | -1 | -1 | -1 |
| N | -1 | 19 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 20 | -1 | -1 | -1 | -1 |
| P | -1 | -1 | -1 | 18 | -1 | -1 | 18 | 18 | 16 | 17 | -1 | 18 | -1 | -1 | -1 |
| Y | -1 | -1 | 3 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 2 | -1 |

说明：

1. 第1列为终结符，第1行为非终结符

2. 非终结符与终结符的交点出表示将要选择哪个产生式做预测分析进行推导

3. -1表示出错，其余数字为产生式的序号NO，表示选择相应的产生式

4.6文法分析

本次设计的文法是：

S1->for(W)Sx

W->P;W1;W2

W1->iOi

W2->iO1

O->>|<|>=|<=

O1->++|--

Sx->{Am}

Am->PAm1

Am1->;PAm1|kong

P->i=E

E->TE1

E1->+TE1 |-TE1 | kong

T->FT1

T1->\*FT1 |/FT1| kong

F->(E) | i |digit

# 中间代码形式

以for(i=0;i<n;i=i+1){m=(n-1)/5-3\*2;}为例：

中间代码序列的结构应如下：

1. i=0

2. if i<n goto 4

3. goto 11

4. t1=n-1

5. t2=t1/5

6. t3=3\*2

7. t4=t2-t3

8. m=t4

9. i=i+1

10. goto 2

11.

我的中间代码结构是这样设计的：

第1条语句输出表达式1，表达式为空时输出空行；

第2条输出条件判断；

第3条输出goto temp+7；

第4条至第4+temp条输出赋值语句的四元式；

第5+temp条输出条件表达式3；

第6+temp条输出"goto 2"

第7+temp条输出空

说明：其中temp表示最终生成的中间变量个数，例如上例中，生成了t1，t2,t3,t4共4个中间变量，所以temp=4.

由于临时变量的产生个数是未知的，在语法制导翻译中要先对语句进行保存，等临时变量temp产生完后在对相应中间代码语句进行回填，最后输出。以便达到输出的中间代码跳转出入口及顺序是正确的。

# 系统测试

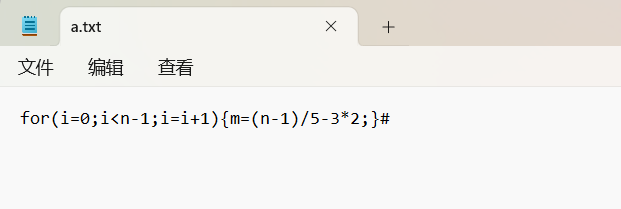
## 测试方法

根据自己当初的设计思想，设计了测试用例。包括没有for中第1个表达式的情况，没有第3个表达式的情况，没有赋值语句的情况，赋值语句比较复杂等情况。用文本输入for循环语句，用txt文本输出分析出的四元式。

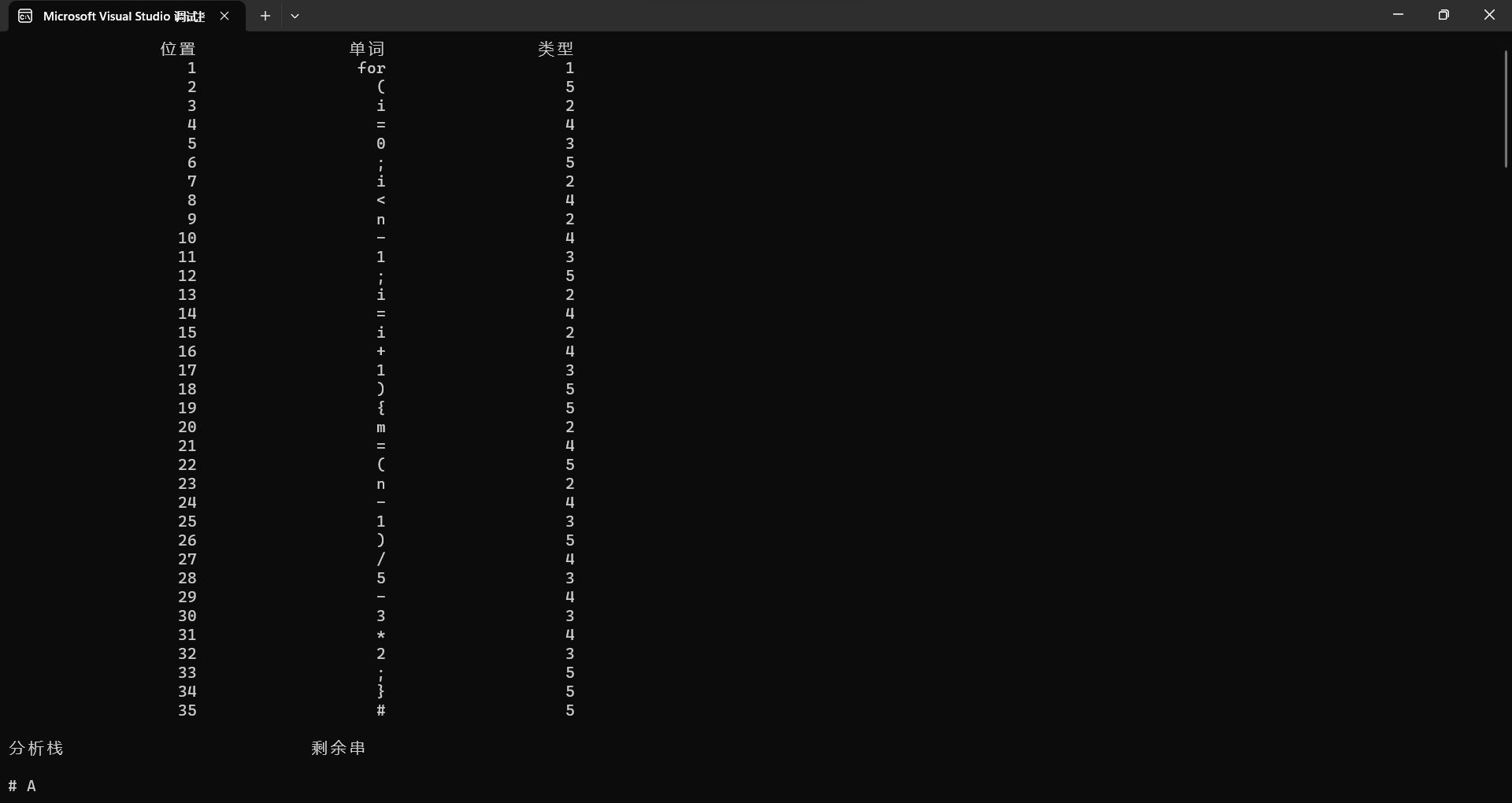
## 测试用例

在文本文件中输入源程序： for(i=0;i<n;i=i+1){m=(n-1)/5-3\*2;}

输入测试文件：



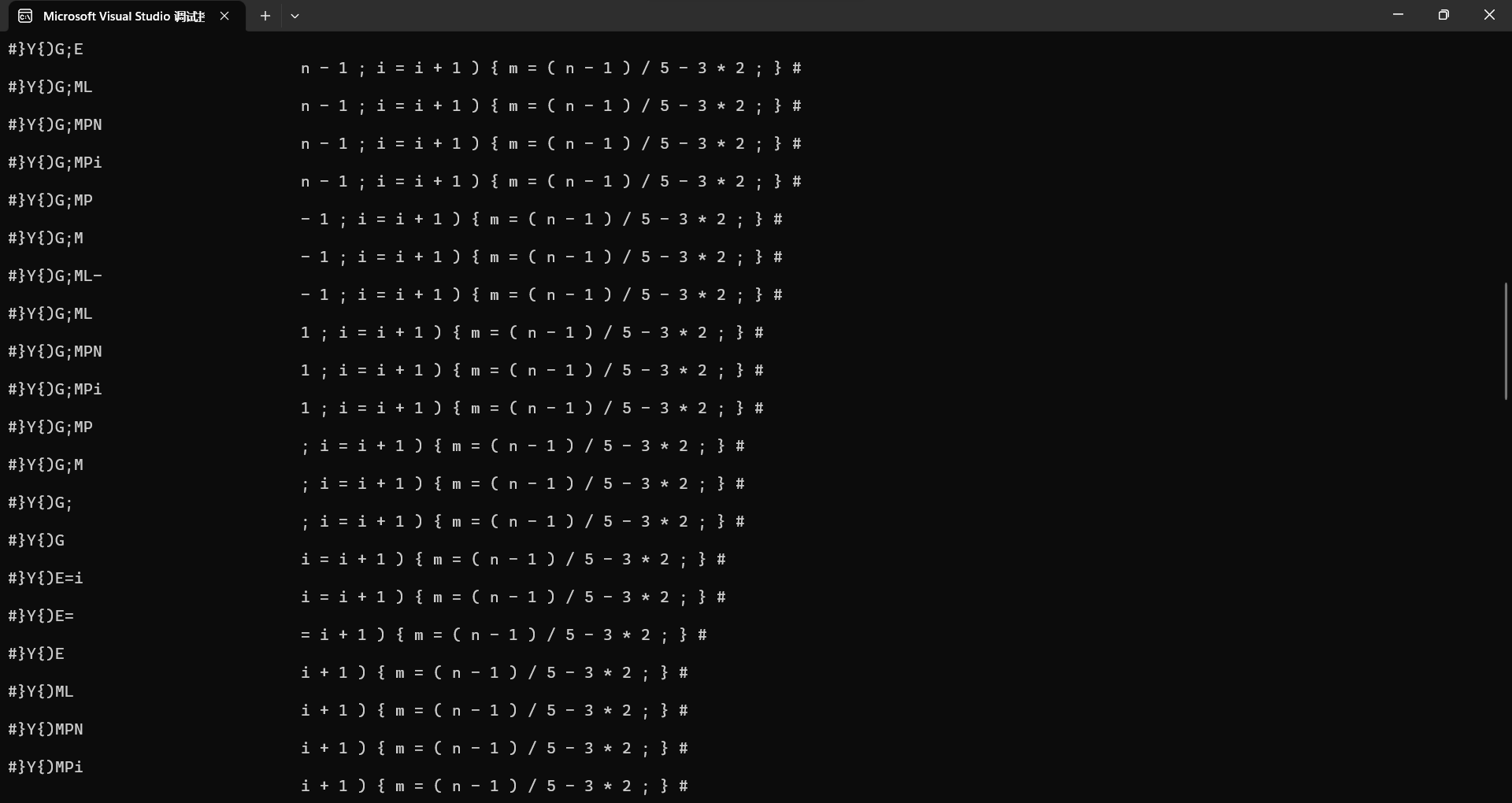
词法分析结果：

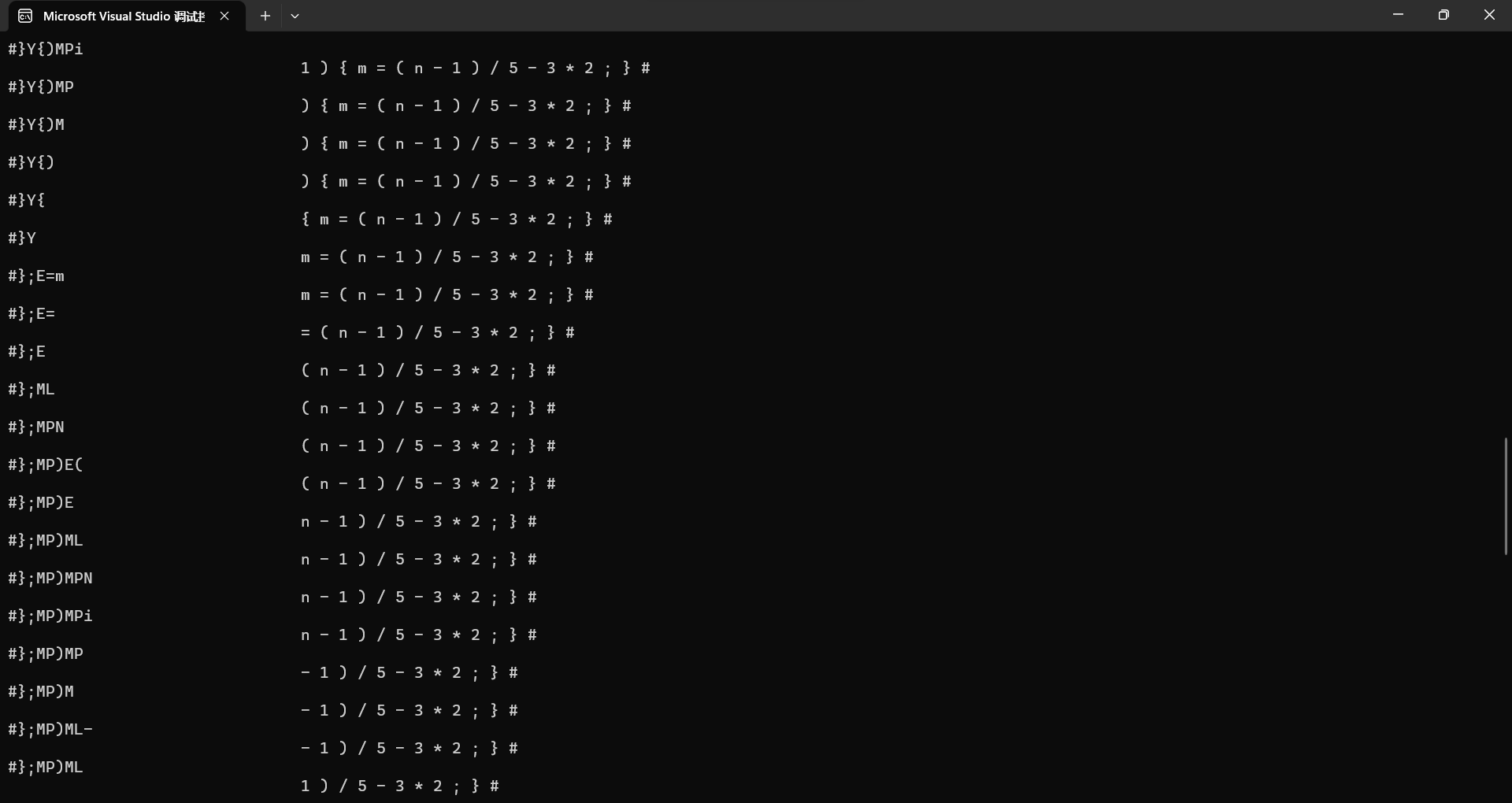


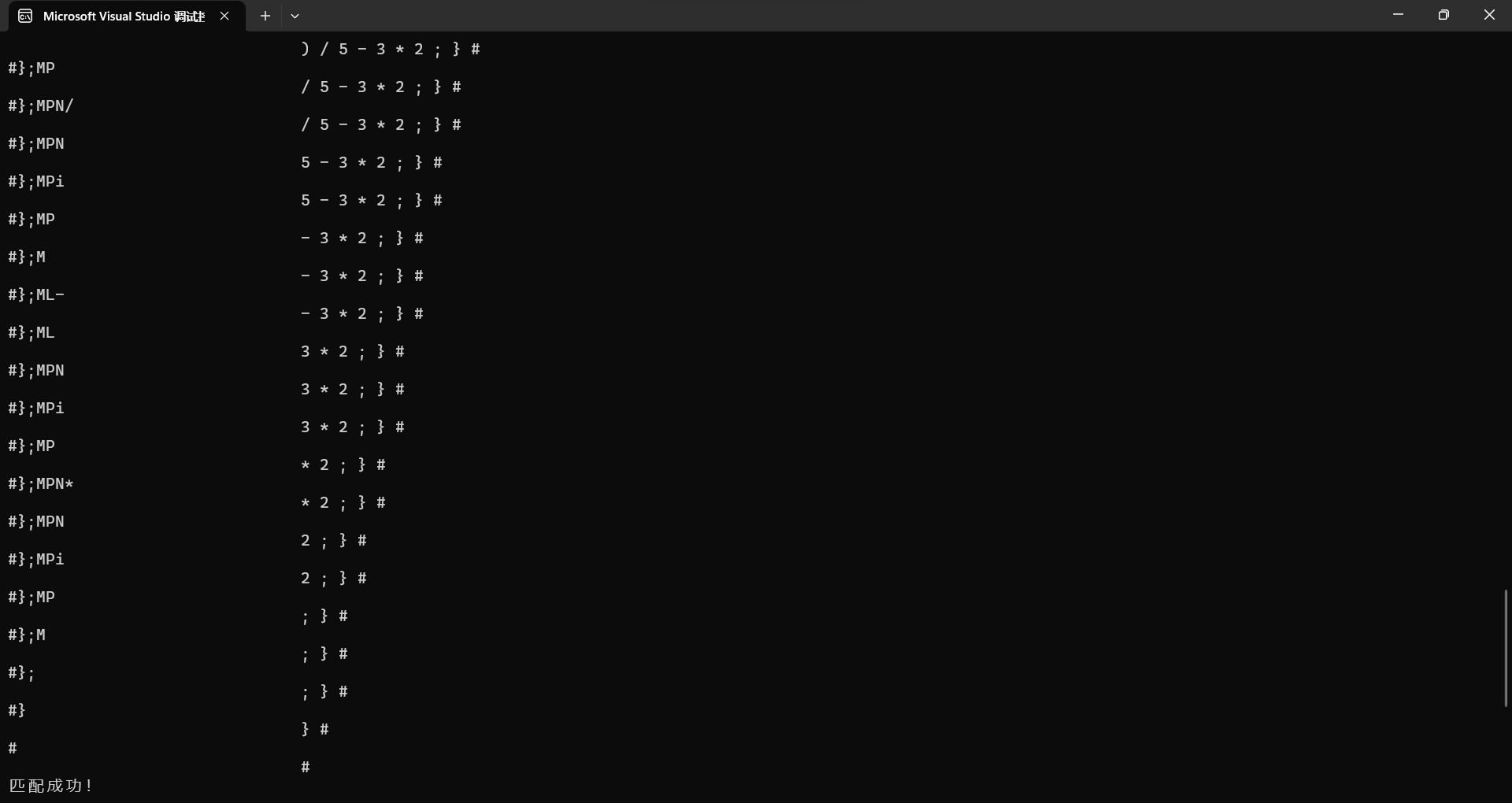
其中第一列为位置序号；第二列为截取的单词；第三列为类型

类型1：关键字 类型2：标识符 类型3：常数 类型4: 运算符 类型5：界符

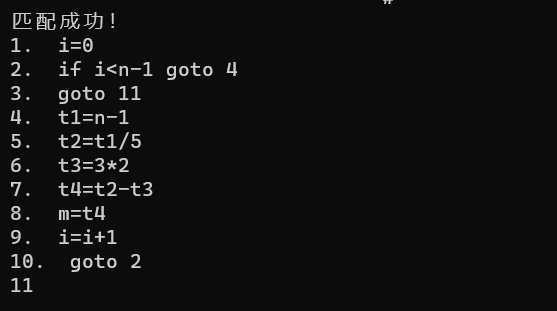
语法分析结果：



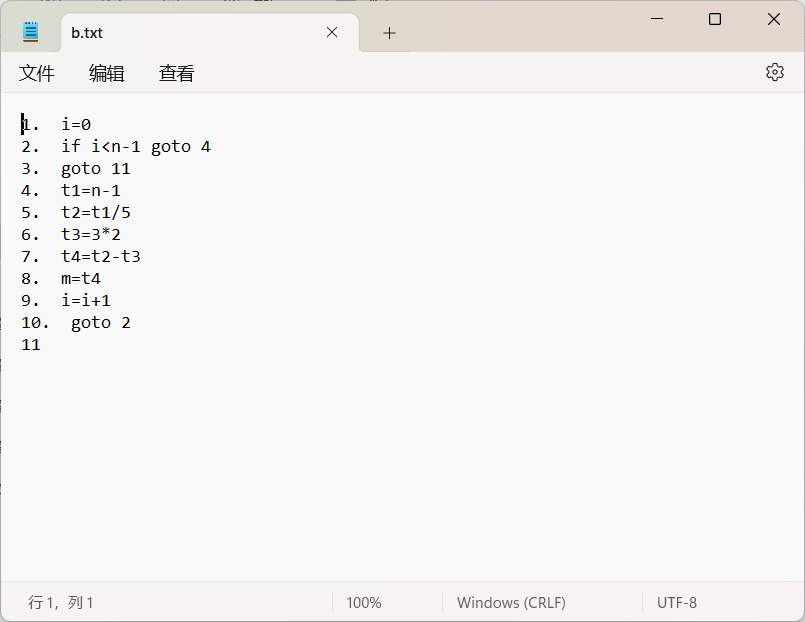




语义分析结果：

****

将四元式进行输出，txt文档记录：



# 心得体会

#### 学生A

这次的实验是让我们通过之前实验的内容，以及上课所学习的知识，自己做一个完整的编译器，来进行分析，从输入源代码-词法分析-语法分析-汇编代码生成的整个过程，都有我们自己完成并且衔接，也就是我们前几次实验的总体整合。但是在这之上我们又添加LR文法编译程序，将LR文法表通过Python程序编译成C++代码。

我们一起共同制定了符号表，并搭建了框架。而我的任务主要是符号表的整理，以及词法分析部分的实现。还有PPT制作以及课程设计报告的撰写。设计出一款程序设计语言的编译器，可以看作是一个程序员编程能力进步的一个里程碑一样的事件，因此这对我们来说也是一种非常特别的挑战。期间还出现了一点点的小插曲。我们在测试第一次编写的LR文法表的时候，发现经常出现问题，后来又重新计算了一遍文法表，发现原来是第一次计算错了，几个小时就这样白白浪费了。其实说起来也没有浪费，从中我们学习到了认真的重要性，要及时做review，不然底子不扎实，顶层就不会好到哪去。摩天大厦就建不成。

在课程设计的五天内，我们小组重温了词法分析器，语法分析器等相关知识与概念，查找了x86汇编语言的语法规范，同时尝试了为程序调入参数以从命令行调用，并且在调试的过程中又对整体有了更明确的体会。在课程设计之后，我们也会更加全力的优化代码，为这个“嘻咖咖”的编译器增加更多强大的功能。

#### 学生B

通过课程设计，不光代码和理论方面得到了提升同时也在其他认知层面有了新的认识。到现在我才明白，原来的每一次实验都是为了这次的课程设计安排的。从最开始的词法解析器，再到自顶向下的文法解析器，以及LL（1）文法解析器。再到后来的四元式生成。每一次的实验我都认真的完成了，很清晰每次都做了什么并且奠定了很扎实的基础。

本次课设就很好的利用了以前学过的知识，并稍加改造。将以前的文法分析子程序封装成了一个Lexer对象，然后改造成迭代器。在后面其他模块的使用中方便了许多。其中也不乏有很多的惊喜。根据陈元绘制的LR分析器的流程图，一次性的编写了LR分析器，然后很惊喜的是，编写代码过程中没有进行任何的调试和运行，但是在第一次运行的时候就直接成功了。当时的喜悦无法用文字来形容，多巴胺剧烈的分泌着，极度愉悦的心情促使我完成了剩下的属于我的任务。但是后面有几个小失败也打击了我不少的自信心。

总之就是收获了很多，但是也辛苦了很久，但是依然很快乐。

#### 学生C

通过本次课程设计，我和我的组员共同努力，一起编写了一个“嘻咖咖”编译器，掌握了什么是编译程序，编译程序工作的基本过程及其各阶段的基本任务，熟悉了编译程序总流程框图，了解了编译程序的生成过程、构造工具及其相关的技术对课本上的知识有了更深的理解。

通过把该算法的内容，算法的执行顺序在计算机上实现，把原来以为很深奥的书本知识变的更为简单，对实验原理有更深的理解。本学期为期一周的嵌入式课程设计在不知不觉中结束了，虽说这次课程设计时间不是很长，但是感觉自己收获颇丰，不仅学习到了一些新知识，回顾了以前的一些快要遗忘的知识点，而且使自己的学习目标更加明确，学习方法更加完善，也体会到软件开发的趣味，更加清楚地认识到了自己在软件开发及学习上的一些不足之处。

在进行这门课程之前，我都以为编译器是天造之物，他本来就是存在的，因为他太完美了，简直不像是出自我们这些看起来笨笨拙拙的大学生之手。但是学了这门课，我，悟了。编译都是基于这些自动机来完成，编译原理就是集各大算法于大成的一个高级程序。她就像一个神奇的机器一样，但是现在她也不再神秘了。无论是词法分析还是LR分析器，我现在都很清楚他们是什么了，他们也不止在编译原理里面能够应用的到，我想，在任何能够处理文本的地方也可以用的到吧。

学完了这门课不止让我学会了编译器，还让我开拓了思路，编程的范式又增加了。

# 8课程设计日志

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 工作内容 | 成员分工 | | |
| 学号 | 学号 | 学号 |
| 学生A | 学生B | 学生C |
| 6.12 | 准备编译原理预备知识  基本框架的讨论，具体实现的功能，确定了最近几天大致的分工。 | 语义分析、四元式生成、熟悉了构造对应分析程序的相关原理，理清整体编写程序的思路。 | 符号表的整理、词法分析、熟悉了构造对应分析程序的相关原理，理清整体编写程序的思路。 | 文法语法分析、绘制流程图、熟悉了构造对应分析程序的相关原理，理清整体编写程序的思路。 |
| 6.13 | 把设计好的文法绘制出该文法的DFA框图。框图化简，求Follow集合，用SLR(1)分析表的构建方法完成LR分析表。 | 继续进行语义分析、四元式生成 | 继续进行符号表的整理、词法分析 | 继续进行文法语法分析、绘制流程图 |
| 6.14 | 开始编写词法分析器。然后编写文法分析器等等。 | 继续进行语义分析、四元式生成 | 继续进行符号表的整理、词法分析 | 继续进行文法语法分析、绘制流程图 |
| 6.15 | 完善报告，准备汇报。优化了实验代码，解决LR文法推导错误，大家一起准备好了答辩材料。 | 编辑了课程设计报告，重新推导LR分析表 | 对PPT进行了整理 | 整理了流程图等内容 |
| 6.16 | 进行了汇报与展示，老师对我们提出了新的问题 | 准备/答辩 | 准备/答辩 | 准备/答辩 |

**指导教师评分**：（小组总评）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 评定项目 | | 内容 | 满分 | 评分 | 课程目标得分 | 总分 |
| 课程目标1（40%） | 课程设计报告（100%）  需求分析文档规范 | 总结系统需求。需求分析文档规范、合理。 | 40 |  |  |  |
| 课程目标2（30%） | 课程设计日志（40%） | 能够查阅和利用文献，能够积极有效参与讨论，学习认真，态度端正，遵守纪律，具备团队沟通和协作能力。 | 12 |  |  |
| 课程设计验收（60%）  程序演示 | 能够根据项目需求分析，提出项目正确的设计方案，完成系统设计、代码设计，系统实施，系统测试和验证。代码规范；系统界面友好。能够合理扩展系统功能、改善性能。 | 18 |  |
| 课程目标3（30%） | 课程设计验收（40%）  答辩情况 | 汇报简明扼要，思路清晰；表达准确。回答问题有理有据，基本原理清楚。主要问题回答准确，有一定深度。 | 12 |  |  |
| 课程设计报告（60%）  小组总结 | 报告内容完整、逻辑关系清楚、语言表达流畅、有一定的个人见解，没有抄袭现象，上交及时。 | 18 |  |
| 总成绩 | 采用五级分制：优、良、中、及格、不及格 | | | | |  |
| 评 语： | | | | | | |
| 指导教师： 日期： | | | | | | |

组长成绩：学号： 、 姓名： 学生A

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 评定项目 | | 内容 | 满分 | 评分 | 课程目标得分 | 总分 |
| 课程目标1（40%） | 课程设计报告（100%）  需求分析文档规范 | 总结系统需求。需求分析文档规范、合理。 | 40 |  |  |  |
| 课程目标2（30%） | 课程设计日志（40%） | 能够查阅和利用文献，能够积极有效参与讨论，学习认真，态度端正，遵守纪律，具备团队沟通和协作能力。 | 12 |  |  |
| 课程设计验收（60%）  程序演示 | 能够根据项目需求分析，提出项目正确的设计方案，完成系统设计、代码设计，系统实施，系统测试和验证。代码规范；系统界面友好。能够合理扩展系统功能、改善性能。 | 18 |  |
| 课程目标3（30%） | 课程设计验收（40%）  答辩情况 | 汇报简明扼要，思路清晰；表达准确。回答问题有理有据，基本原理清楚。主要问题回答准确，有一定深度。 | 12 |  |  |
| 课程设计报告（60%）  小组总结 | 报告内容完整、逻辑关系清楚、语言表达流畅、有一定的个人见解，没有抄袭现象，上交及时。 | 18 |  |
| 总成绩 | 采用五级分制：优、良、中、及格、不及格 | | | | |  |

组员1成绩： 学号： 、 姓名： 学生B

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 评定项目 | | 内容 | 满分 | 评分 | 课程目标得分 | 总分 |
| 课程目标1（40%） | 课程设计报告（100%）  需求分析文档规范 | 总结系统需求。需求分析文档规范、合理。 | 40 |  |  |  |
| 课程目标2（30%） | 课程设计日志（40%） | 能够查阅和利用文献，能够积极有效参与讨论，学习认真，态度端正，遵守纪律，具备团队沟通和协作能力。 | 12 |  |  |
| 课程设计验收（60%）  程序演示 | 能够根据项目需求分析，提出项目正确的设计方案，完成系统设计、代码设计，系统实施，系统测试和验证。代码规范；系统界面友好。能够合理扩展系统功能、改善性能。 | 18 |  |
| 课程目标3（30%） | 课程设计验收（40%）  答辩情况 | 汇报简明扼要，思路清晰；表达准确。回答问题有理有据，基本原理清楚。主要问题回答准确，有一定深度。 | 12 |  |  |
| 课程设计报告（60%）  小组总结 | 报告内容完整、逻辑关系清楚、语言表达流畅、有一定的个人见解，没有抄袭现象，上交及时。 | 18 |  |
| 总成绩 | 采用五级分制：优、良、中、及格、不及格 | | | | |  |