****

**编译原理课程设计报告**

班 级： 计科152 \_

学 号： 150730 \_

姓 名： \_

指导老师： 黄汝维 \_

目录

[一、需求分析 3](#_Toc535600539)

[二、环境说明 3](#_Toc535600540)

[三、详细设计 3](#_Toc535600541)

[四、调试分析 8](#_Toc535600542)

[五、课程总结 12](#_Toc535600543)

## 一、需求分析

根据组内分供，我主要负责编写语法分析部分，故在此报告中将着重对语法分析部分进行阐述。语法分析是根据某种给定的形式文法对由单词序列(如英语单词序列)构成的输入文本进行分析并确定其语法结构的一种过程。它通常用词法分析所产生的属性表序列作为输入,来识其输入的序列是否满足文法的要求。

在本次课程设计中，我们小组决定采用LL(1)分析法来进行语法分析。我们先手动求出文法的select集合，然后得到一个LL1(1)分析表。在语法分析模块中，通过与获得的分析表进行对比，进行入栈出栈操作，匹配到相同的则认为匹配成功，否则报错；直到分析栈和余留字符串都为空，才能算程序通过了语法分析。

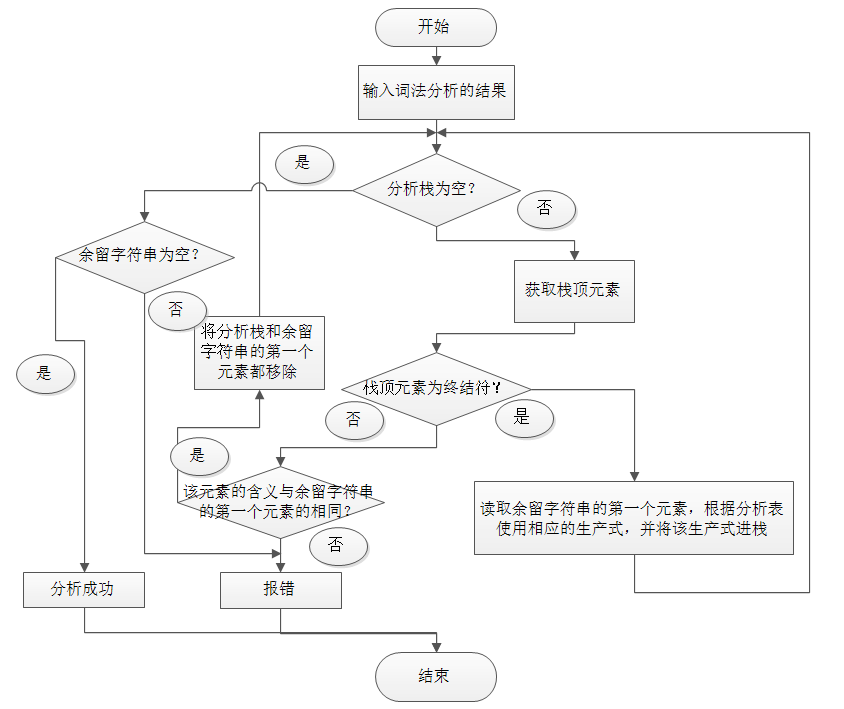
## 二、环境说明

操作系统：Windows系统

开发工具：Eclipse软件

## 三、详细设计

语法分析流程图：



数据结构和存储结构：

AnalyseNode{

String type;//节点类型

String name;//节点名

String value;//节点值

}

类AnalyseNode用于存储单个分析节点，每一个非终结符和终结符就是一个分析节点。语法分析模块中，使用栈来存放分析过程中产生的AnalyseNode，而词法分析生成的结果则使用List来存放。

文法：

S->void main(){A}

A->CA

A->空

C->XBR

/\*声明变量\*/

X->YZ;

X-空

Y->int

Y->char

Y->bool

Z->UZ'

Z'->,Z

Z'->空

/\*赋值语句\*/

U->ASS\_U id U'

U'->L EQ\_U

U'->空

/\*控制语句\*/

B->if(G) IF\_FJ {A} IF\_BACKPATCH\_FJ IF\_RJ else{A} IF\_BACKPATCH\_RJ

B->while(G) WHILE\_FJ {A} WHILE\_RJ WHILE\_BACKPATCH\_FJ

B->for(YZ; G FOR\_FJ ;Q){A SINGLE} FOR\_RJ FOR\_BACKPATH\_FJ

B->scanf("%d",&F)SCANF A;

B->prinf("%d",F)PRINT A;

B->空

/\*赋值语句\*/

R->@ASS\_R id = L @EQ;

R->空

/\*判断语句\*/

G->FDF COMPARE

G->(E)

G->!E

E->HE'

E->&&E

E->空

F->ASS\_F id

F->ASS\_F num

F->(L) TRAN\_LF

F->空

D->COMPARE\_OP ==

D->COMPARE\_OP !=

D->COMPARE\_OP >

D->COMPARE\_OP <

D->COMPARE\_OP >=

D->COMPARE\_OP <=

H->GH'

H->||E

H'->空

L->TL' ADD\_SUB

L'->+L ADD

L'->-L SUB

L'->空

T->FT' DIV\_MUL

T'->/T DIV

T'->\*T MUL

T'->空

/\*自增自减赋值语句\*/

Q->@ASS\_Q id O

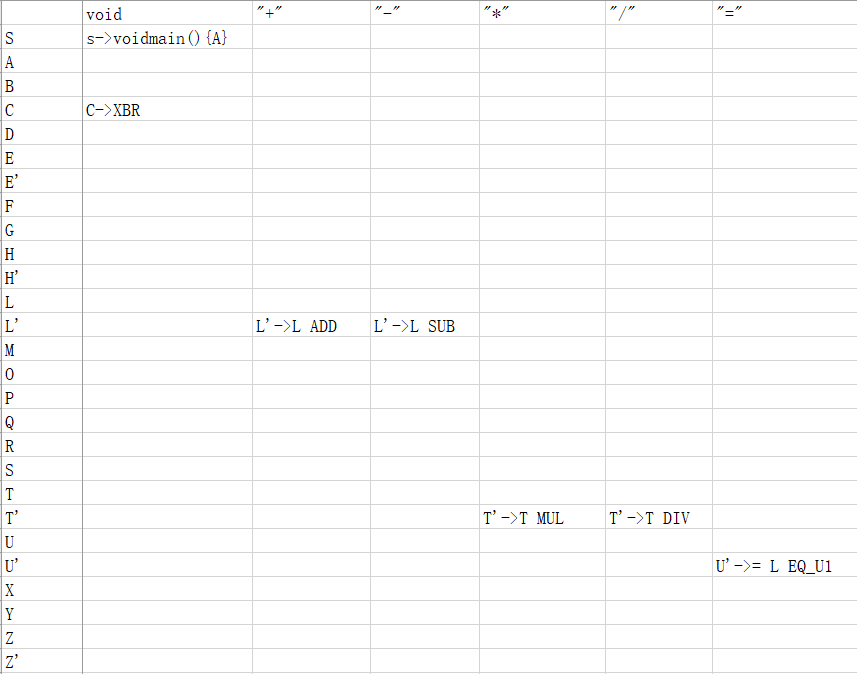
Q->空

O->@SINGLE\_OP ++

O->@SINGLE\_OP --

O->空

分析表部分截图：



## 四、调试分析

测试数据：

void main(){

int i,sum,n;

i=0;

scanf("%d",&n);

if(n<1){

sum=0;

}else{

sum=1;

}

while(i<n){

i=i+1;

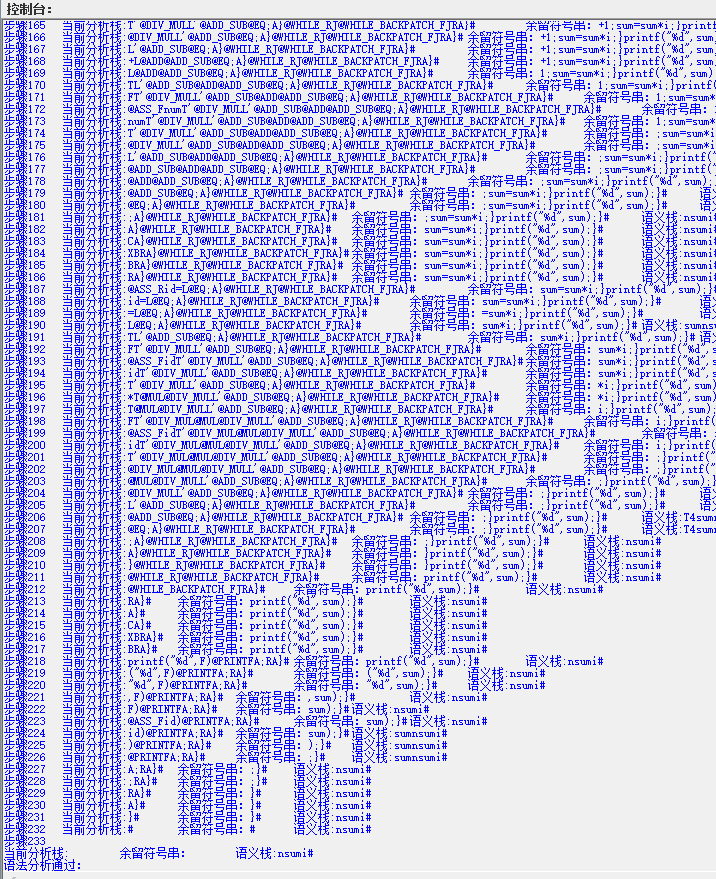
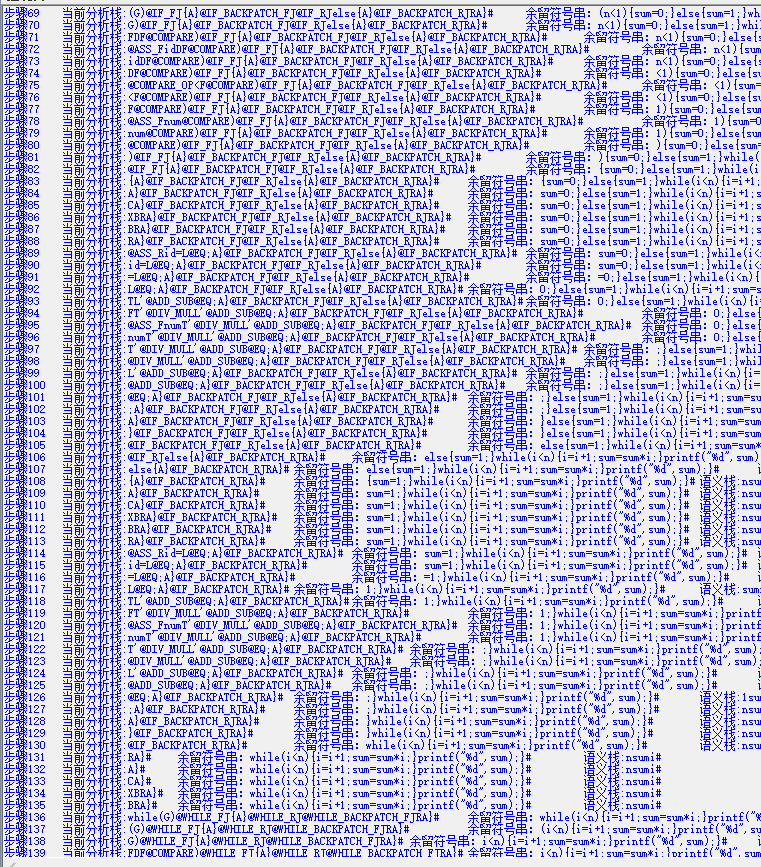
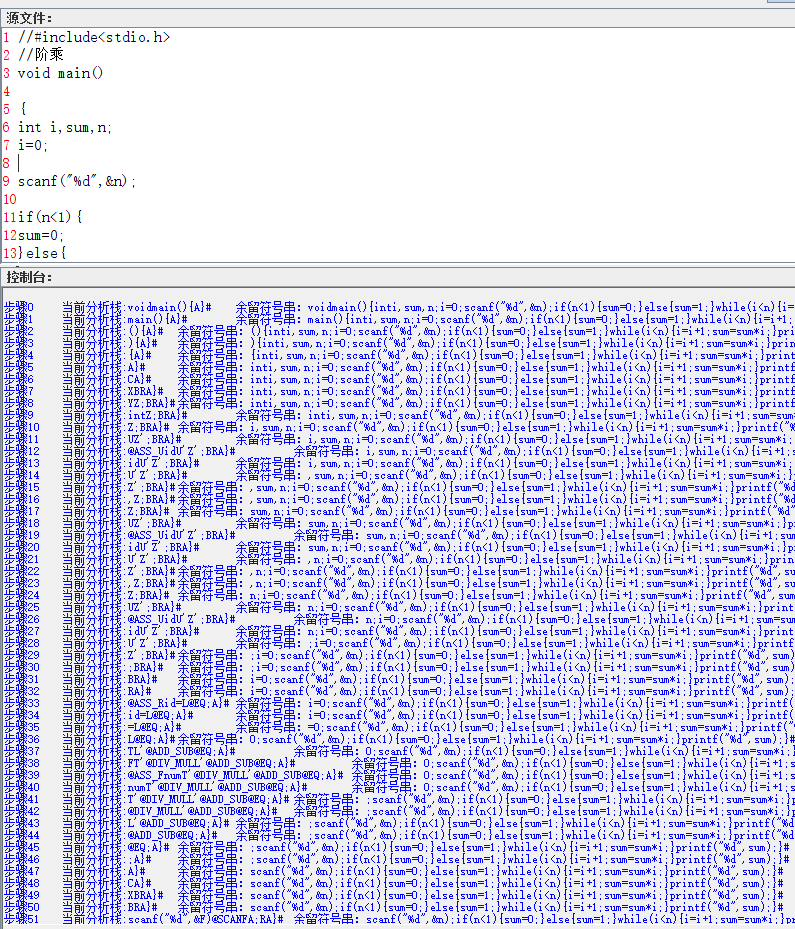
sum=sum\*i;

}

printf("%d",sum);

}

测试结果：



设计时的难题：

由于文法的产生式比较多，不像平时做作业时那么少，所以消除左递归和求select集合比较困难。因此，我们组员都独自求一遍select集合，然后对比每个人之间有什么不同的地方，再一起讨论，以求出正确的select集合。

算法的改进设想：

在语法分析模块中的select集合和分析表都是我们手动求出来的，如果修改了文法中的某些部分，就需要再次手动求出select集合和分析表，然后再修改语法分析模块的程序，十分地麻烦。我们可以在程序中增加一个模块，根据输入消除左递归的文法，自动求出select集合，并将其以（key：非终结符+终结符，value：产生式部）的格式放入Map中。语法分析模块中，根据key得到相应的value，进行入栈出栈操作，而不需要再使用很多个if-else语句。

## 五、课程总结

编译原理的核心是把我们编写的高级语言通过一定的算法转换生成机器执行的二进制代码。学习编译原理可以让我们对语法和数据结构有一个更深层次的认识，可以让我们理解机器是如何理解高级语言的，我们可以学习其中的思想运用到我们构建编写代码过程中，减少错误代码的出现以及能快速写出整洁的代码，也可以方便我们进行代码优化。

这次课程设计不仅仅是让我们把所学的理论知识与实践相结合起来,提高自己的实际动手能力和独立思考的能力,也让我学会从各方意见和想法中提取精华,这次设计中的不少地方都是源于和同学的大量讨论获得的,这种讨论对思维的激发和创新让我收获颇多。这次课程设计使自己学到了不少知识,也经历了不少艰辛, 但收获同样巨大。通过这次课程设计我也发现了自身存在的不足之处, 虽然感觉理论上已经掌握,但在运用到实践的过程中仍有意想不到的困惑,经过一番努力才得以解决。通过这次设计,我懂得了学习的重要性,了解到理论知识与实践相结合的重要意义,学会了坚持、耐心和努力,这将为自己今后的学习和工作做出了最好的榜样, 更重要的是把自己平时所学的东西应用到实际中。其实课程设计反映的是一个从理论到实际应用的过程,它将对我们今后的工作有着深远的影响。让我们学会主动参与学习,主动了解,认识自己的不足,将理论活学活用才能提高自己的能力和竞争力,我会谨记这些收获,不断提高和进步!