

2013 级《编译原理》研究性学习实验教学内容

一、学习目标

编译系统是计算机系统重要的系统软件之一，编译原理课程主要讲授编译系统重要组成部分程序设计语言编译程序的设计原理和实现技术，是计算机科学与技术专业学生的必修专业课程，也是每个优秀的计算机专业人员必修的一门课程，更重要的是，编译原理课程中蕴含着计算机科学中解决问题的思路、抽象问题和解决问题的方法，其内容可让计算机专业学生“享用一辈子”。

本课程开展研究性教学的目的是，在教师的引导下以问题回溯与思维启发的方式，使学生在不断的探究过程中掌握编译程序设计和构造的基本原理和实现技术，启迪学生的抽象思维、激发学生的学习兴趣、培养学生的探究精神和专业素养，从而提高学生发现问题、分析问题和解决问题的能力。

二、学习任务

1、学习内容

以理论与实践紧密结合的六个研究性学习专题作为教学载体，这六个研究专题是：

专题 1_词法分析程序构造原理与实现

专题 2_递归下降语法分析设计原理与实现

专题 3_LL(1) 语法分析设计原理与实现

专题 4_算符优先语法分析设计原理与实现

专题 5_编译器前端实现（专题 1+专题 2、3、4 选择一个形成一个程序）

专题 6_语义分析及中间代码生成程序设计原理与实现

2、任务要求

1) 以上内容专题 1 必做；专题 2、3、4 选择二个必做；专题 5 必做；专题 6 选做；实验占总成绩的 20%，完成选做部分在总成绩中将获得加分。

2) 实验要求设计并完成所做专题，程序应能查看到词法分析的输出二元式序列，词法分析和语法分析得的输出结果。或者中间代码序列，还能给出简单的错误提示。

3) 编写符合实验语言词法和语法规则的源程序进行测试，须给出尽可能完备的测试用例。

4) 能力培养：深入理解理论对实践的指导作用；基本原理、实现技术和方法的正确运用；编译程序本身的系统性。鼓励同学扩充语言的功能，如增加单词符号的种类，增加赋值语句、条件语句等。

3、需提交的文档要求

1) 对于必做的专题(专题1-专题4),给出分专题【专题i设计(word)】和【源程序】

【专题i设计(word)】包括:(1)程序功能描述;(2)主要数据结构描述;(3)程序结构描述:设计方法、函数定义;(4)程序测试:测试用例、测试结果。

提交文件夹命名方式“学号_姓名_专题i”,内含分专题设计和源程序,词法分析程序提交截止日期2016年4月29日,其它提交截止时间2016年6月17日前。

2) 对专题5,给出【编译器前端设计文档(word)】和【编译器前端源程序】

【编译器前端设计文档(word)】包括:(1)程序总体结构,与分专题的关系,测试用例和测试结果。(2)学习总结:实验过程中遇到的主要问题、如何克服、对你的程序进行评价、对问题的深入理解和实验的收获。

提交文件夹命名方式“学号_姓名_专题5”,内含设计文档和源程序,提交截止时间:2016年6月17日前。

说明:实验报告提交到课程平台。

实验检查集中在第15周和第16周。

三、学习内容设计

■ 研究性学习专题1 词法分析程序构造原理与实现

1、理论传授

源程序输入与词法分析程序输出的基本方法;正则文法及其状态转换图的基本概念,正则表达式及有限自动机的基本概念;正规文法构造相应的状态转换图的基本方法;正则表达式构造有限自动机的基本方法及不确定有限自动机确定化的基本方法;词法分析程序的设计与编写。

2、目标任务

【实验项目】完成以下正则文法所描述的C语言子集单词符号的词法分析程序。

〈标识符〉→字母 | 〈标识符〉字母 | 〈标识符〉数字

〈无符号整数〉→数字 | 〈无符号整数〉数字

〈单字符分界符〉→+ | - | * | ; | , | (|) | { | }

〈双字符分界符〉→〈大于〉= | 〈小于〉= | 〈小于〉> | 〈感叹号〉= | 〈等于〉= | 〈斜竖〉*

〈小于〉>〈 〈等于〉>= 〈大于〉>> 〈斜竖〉>/

〈感叹号〉>!

该语言的保留字: void int float double if else for do while

请自己改写文法加入增量和减量运算符: ++, --, >>, <<, +=, -=, *=, /= 等。

【设计说明】(1)该语言大小写不敏感;(2)字母为a-z A-Z,数字为0-9;(3)可以对

上述文法进行扩充和改造；(4) “/*.....*/” 为程序的注释部分。

[设计要求] (1) 给出各单词符号的类别编码；(2) 词法分析程序应能发现输入串中的错误；(3) 词法分析作为单独一遍编写，词法分析结果为二元式序列组成的中间文件；

[任务分析] 重点解决正则文法到状态转换图的转化问题。

■ 研究性学习专题 2 递归下降语法分析设计原理与实现

1、理论传授

语法分析的设计方法和实现原理；LL(1) 文法及其判定；无回溯的递归下降分析的设计与实现；

2、目标任务

[实验项目] 完成以下描述算术表达式的 LL(1) 文法的递归下降分析程序

G[E]: $E \rightarrow TE'$
 $E' \rightarrow ATE' \mid \varepsilon$
 $T \rightarrow FT'$
 $T' \rightarrow MFT' \mid \varepsilon$
 $F \rightarrow (E) \mid i$
 $A \rightarrow + \mid -$
 $M \rightarrow * \mid /$

[设计说明] 终结符号 i 为用户定义的简单变量, 即标识符的定义。

[设计要求] (1) 输入串应是词法分析的输出二元式序列, 即某算术表达式“专题 1”的输出结果, 输出为输入串是否为该文法定义的算术表达式的判断结果；(2) 递归下降分析程序应能发现输入串出错；(3) 设计两个测试用例 (尽可能完备, 正确和出错), 并给出测试结果。

4、任务分析 重点解决 LL(1) 文法到递归子程序框图的设计。

■ 研究性学习专题 3 LL(1) 语法分析设计原理与实现

1、理论传授

语法分析的设计方法和实现原理；LL(1) 分析表的构造；LL(1) 分析过程；LL(1) 分析器的构造；

2、目标任务

[实验项目] 实现 LL(1) 分析中控制程序 (表驱动程序); 完成以下描述算术表达式的 LL(1) 文法的 LL(1) 分析程序。

G[E]: $E \rightarrow TE'$
 $E' \rightarrow ATE' \mid \varepsilon$
 $T \rightarrow FT'$
 $T' \rightarrow MFT' \mid \varepsilon$

$$F \rightarrow (E) \mid i$$
$$A \rightarrow + \mid -$$
$$M \rightarrow * \mid /$$

说明：终结符号 i 为用户定义的简单变量, 即标识符的定义。

[设计要求] (1) 输入串应是词法分析的输出二元式序列, 即某算术表达式“实验项目一”的输出结果。输出为输入串是否为该文法定义的算术表达式的判断结果; (2) LL(1) 分析过程应能发现输入串出错; (3) 设计两个测试用例 (尽可能完备, 正确和出错), 并给出测试结果。

3、任务分析 重点解决 LL(1) 表的构造和 LL(1) 分析器的实现。

■ 研究性学习专题 4 算符优先语法分析设计原理与实现

1、理论传授

语法分析的设计方法和实现原理, 算符优先文法、最左素短语、算符优先矩阵、优先函数的基本概念; 算符优先文法句型最左素短语的确定; 算符优先分析算法的实现;

2、目标任务

[实验项目] 实现算符优先分析算法, 完成以下描述算术表达式的算符优先文法的算符优先分析过程。

$$G[E]: E \rightarrow E+T \mid E-T \mid$$
$$T \rightarrow T * F \mid T / F \mid F$$
$$F \rightarrow (E) \mid i$$

说明：终结符号 i 为用户定义的简单变量, 即标识符的定义。

[设计要求] (1) 构造该算符优先文法的优先关系矩阵或优先函数; (2) 输入串应是词法分析的输出二元式序列, 即某算术表达式“专题 1”的输出结果。输出为输入串是否为该文法定义的算术表达式的判断结果。(3) 算符优先分析过程应能发现输入串出错。(4) 设计两个测试用例 (尽可能完备, 正确和出错), 并给出测试结果。

3、任务分析 重点解决算符优先矩阵的构造和算符优先算法的实现。

4、能力培养 深入理解理论对实践的指导作用; 基本原理、实现技术和方法的正确运用。

■ 研究性学习专题 5 编译器前端实现

1、理论传授

词法分析的输入和输出, 二元式的表示, 二元式作为语法分析程序的输入, 语法分析程序的输出, 理解编译程序的整体概念, 理解编译器前端的概念。

2、目标任务

[实验项目] 将词法分析程序 (专题 1) 和语法分析程序 (在专题 2、3、4 中选择一个) 形成一个程序, 即程序的输入为符号串源程序 (算术表达式), 输出为判断结果 (该符号串

是否为正确的算术表达式)。

[设计要求] (1) 词法分析的输入为符号串 (算术表达式), 词法分析的输出为二元式形式的中间文件; (2) 语法分析读取二元式形式的中间文件并判断符号串是否为正确的算术表达式; (3) 设计两个测试用例 (尽可能完备), 并给出程序执行结果。

3、任务分析 重点解决编译程序总体结构理解问题; 词法分析的输出为语法分析的输入。

■ 研究性学习专题 6 语法制导翻译程序设计原理与实现技术

1、理论传授

语法制导的基本概念, 目标代码结构分析的基本方法, 赋值语句语法制导生成四元式的基本原理和方法, 该过程包括语法分析和语义分析过程。

2、目标任务

[实验项目] 完成以下描述赋值语句和算术表达式文法的语法制导生成中间代码四元式的过程。 $G[A]: A \rightarrow V := E$

$$\begin{aligned} E &\rightarrow E+T \mid E-T \mid \\ T &\rightarrow T * F \mid T / F \mid F \\ F &\rightarrow (E) \mid i \\ V &\rightarrow i \end{aligned}$$

说明: 终结符号 i 为用户定义的简单变量, 即标识符的定义, SLR (1) 分析表见附件。

[设计要求] (1) 设计语法制导翻译过程, 给出每一产生式对应的语义动作; (2) 设计中间代码四元式的结构 (暂不与符号表有关)。(3) 输入串应是词法分析的输出二元式序列, 即某算术表达式“专题 1”的输出结果。输出为输入串的四元式序列中间文件。(4) 设计两个测试用例 (尽可能完备), 并给出程序执行结果四元式序列。

3、任务分析 重点解决赋值语句文法的改写和语义动作的添加。

文法 $G[A]$ 的 SLR (1) 分析表。

$$\begin{aligned} S &\rightarrow A \quad [1] \\ A &\rightarrow V = E \quad [2] \\ E &\rightarrow E + T \quad [3] \mid T \quad [4] \\ T &\rightarrow T * F \quad [5] \mid F \quad [6] \\ F &\rightarrow (E) \quad [7] \mid i \quad [8] \\ V &\rightarrow i \quad [9] \end{aligned}$$

状态	ACTION							GOTO					
	i	=	+	*	()	#	S	A	V	E	T	F
0	S2						acc		1	3			
1							R1						
2		R9					R9						
3		S4											
4	S9				S8						5	6	7
5			S10				R2						
6			R4	S11		R4	R4						
7			R6	R6		R6	R6						
8	S9				S8						12	6	7
9			R8	R8		R8	R8						
10	S9				S8							13	7
11	S9				S8								14
12			S10			S15							
13			R3	S11		R3	R3						
14			R5	R5		R5	R5						
15			R7			R7	R7						

四、学习考核形式

实验考核采取当面演示和质疑、设计文档检查相结合的方式。