**编译原理课程实验报告**

**实验2：语法分析**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 张志路 | | 院系 | | 计算机学院 | | | 学号 | | | 1160300909 | | |
| 任课教师 | | 辛明影 | | | | 指导教师 | 辛明影 | | | | | | |
| 实验地点 | | 格物208 | | | | 实验时间 | 2019年4月21日周日5-6节 | | | | | | |
| 实验课表现 | | 出勤、表现得分 | |  | | 实验报告  得分 |  | | 实验总分 | | |  | |
| 操作结果得分 | |  | |
| **一、需求分析** | | | | | | | | | | 得分 | | |  |
| 要求：采用至少一种句法分析技术（SLR(1)、LR(1)）对类高级语言中的基本语句进行句法分析。阐述句法分析系统所要完成的功能。  在词法分析器的基础上设计实现类高级语言的语法分析器，基本功能如下：   1. 能识别以下几类语句： 2. 声明语句（变量声明） 3. 表达式及赋值语句（简单赋值） 4. 分支语句：if\_then\_else 5. 循环语句：do\_while 6. 要求编写自动计算CLOSURE(I)和GOTO函数的程序，并自动生成LR分析表。 7. 具备简单语法错误处理能力，能准确给出错误所在位置，并采用可行的错误恢复策略。输出的错误提示信息格式如下：Error at Line [行号]：[说明文字] 8. 系统的输入形式：要求可以通过文件导入文法和测试用例,测试用例要涵盖“实验内容”第1条中列出的各种类型的语句，并设置一些语法错误。 9. 系统的输出分为两部分：一部分是打印输出语法分析器的LR分析表。另一部分是打印输出语法分析结果，既输出归约时的产生式序列。 | | | | | | | | | | | | | |
| **二、文法设计** | | | | | | | | | | 得分 | | |  |
| 要求：给出如下语言成分的文法描述。  本文法在实验指导书给定的文法基础上做了相应的改进，消除了二义性。   1. 全局定义   P' → P  P → D  P → S  S → S S   1. 声明语句（变量声明**）**   D → D D | proc id ; D S | T id ;  T → X C | record D end  X → integer | real  C → [ num ] C |   1. 表达式及赋值语句   S → id = E ; | L = E ;  E → E + E1 | E1  E1 → E1 \* E2 | E2  E2 → ( E ) | - E | id | num | L  L → id [ E ] | L [ E ]   1. 分支语句“if\_then\_else”和循环语句“do\_while”   S → S1 | S2  S1 → if B then S1 else S1 | while B do S0  S2 → if B then S1 else S2 | if B then S0  S0 → begin S3 end  S1 → begin S3 end  S2 → begin S3 end  S3 → S3 ; S | S   1. 布尔表达式   B → B or B1 | B1  B1 → B1 and B2 | B2  B2 → not B | ( B ) | E R E | true | false  R → < | <= | == | != | > | >=   1. 过程调用   S → call id ( EL )  EL → EL , E  EL → E | | | | | | | | | | | | | |
| **三、系统设计** | | | | | | | | | | 得分 | | |  |
| 要求：分为系统概要设计和系统详细设计。   1. 系统概要设计：给出必要的系统宏观层面设计图，如系统框架图、数据流图、功能模块结构图等以及相应的文字说明。   (1)系统框架图    系统分为基础层、逻辑层、数据层和显示层，基础层即在Windows10、jdk1.8环境下利用java语言进行编程，逻辑层即词法分析程序和语法分析程序，数据层包含测试样例、Token序列、文法和LR分析表，显示层显示一个主界面。  (2) 数据流图    语法分析程序通过读入文法，生成LR分析表，然后读入对测试样例的词法分析结果，根据分析表对其进行LR(1)分析，从而生成规约的产生式序列和错误报告。  (3) 功能模块结构图    如上图所示，语法分析程序的功能主要为读入数据、分析数据和给出结果。对于数据处理模块，首先要设计合理的数据结构，对文法进行存储和表示；然后根据文法生成状态集和转移函数，进而构造分析表；最后对词法分析得到的Token序列利用分析表进行移进规约操作，直到分析完成。   1. 系统详细设计：对如下工作进行展开描述 2. 核心数据结构的设计 3. **Production类**     Production类对一个普通的产生式进行表示，此时表示的是类似于“A → BCD”的形式。   1. **ProductionState类**     ProductionState类对一个产生式状态进行表示，此时表示类似于“A → BC**.**D, a”的形式。   1. **DFAState类**     DFAState类用来表示LR(1)分析的一个DFA状态，即一个项目集闭包，id为项目集编号。   1. **DFAStateSet类**     DFAState类用来表示LR(1)分析法的所有DFA状态，即所有项目集合。   1. **GrammarProc类**     GrammarProc类用来读入并存储类似于“A → BC | DE”形式的产生式，根据前述的数据结构进行存储，产生终结符、非终结符以及每个符号的FIRST集。   1. **AnalyzeTable类**   AnalyzeTable类用来构造一个分析表，此类构造DFA状态集和转移函数，进而构造LR(1)分析表。  主要数据结构如下图所示。     1. **SyntaxParser类**   SyntaxParser类根据构造的语法分析表进行LR(1)语法分析。     1. 主要功能函数说明 2. **求FIRST集**   不断应用下列规则，直到没有新的终结符或ε可以被加入到任何FIRST集合中为止。   1. 如果X是一个终结符，那么FIRST(X)={X}。 2. 如果X是一个非终结符，且 X→Y1…Yk∈P (k≥1)，那么，    1. 如果对于某个i，a在FIRST (Yi)中，且在所有的FIRST(Y1) , … , FIRST(Yi-1)中，就把a加入到FIRST(X)中。    2. 如果对于所有的j=1,2, . . . , k，在FIRST(Yj)中，那么将加入到FIRST(X)。 3. 如果 X→∈P，那么将加入到FIRST( X )中。 4. **求项目集闭包**   根据下列式子构造项目集闭包。  CLOSURE(I) = I∪{[B→.,b] | [A→.B,a]∈CLOSURE(I), B→∈P, b∈FIRST(a)}  伪代码如下所示。   1. CLOSURE(I) 2. { 3. repeat 4. for (I中的每个项[A→α.Bβ, a]) 5. for (G'的每个产生式B→γ) 6. for (FIRST(βa)中的每个符号b) 7. 将[B→.γ, b]加入到集合I中； 8. until 不能向I中加入更多的项； 9. until I ； 10. } 11. **求GOTO函数**   根据下列式子构造GOTO函数。  GOTO( I, X ) = CLOSURE({ [A→αX.β, a] | [A→α.Xβ, a]∈I })  伪代码如下所示。   1. GOTO(I, X) 2. { 3. 将J初始化为空集； 4. **for**(I中的每个项[A→α·Xβ, a]) 5. 将项[A→αX·β, a]加入到集合J中； 6. **return** CLOSURE(J)； 7. } 8. **为文法G’构造LR(1)项集族**   伪代码如下所示。   1. items(G') 2. { 3. 将C初始化为{CLOSURE({[S'→.S, #]})}； 4. repeat 5. **for**(C中的每个项集I) 6. **for**(每个文法符号X) 7. **if**(GOTO(I, X)非空且不在C中) 8. 将GOTO(I, X)加入C中； 9. until不再有新的项集加入到C中； 10. } 11. **构造LR(1)语法分析表**   伪代码如下所示。   1. 构造G'的规范LR(1)项集族C={I,I,…,I} 2. 根据I构造得到状态i。状态i的语法分析动作按照下面的方法决定： 3. **If** [A→α·aβ,b]∈I and GOTO(I,a)=I 4. ACTION[i,a]=sj 5. **If** [A→α·Bβ,b]∈I and GOTO(I,B)=I 6. GOTO[i,B]=j 7. **If** [A→α·,a]∈I and A!=S' 8. ACTION[i,a]=rj（j是产生式A→α的编号） 9. **If** [S'→S·,#]∈I 10. ACTION[i,#]=acc; 11. 没有定义的所有条目都设置为“error” 12. **错误处理**   当栈顶状态为s，当前输入符号为a，查表得Action[s,a]=error，这表明语法出错了。  错误处理伪代码如下。   1. 向栈内搜索非终结符A，它的归约前状态为S，将A前面的符号全部弹出栈。 2. 不断地读入输入符号，直到读到a∈Follow(A)集为止， 3. 将**goto**[s,A]压入栈   整个过程就相当于完成了一次归约，只不过归约的不是句柄。这样处理完后，就相当于从一个新的句子成分开始分析。   1. 程序核心部分的程序流程图 | | | | | | | | | | | | | |
| **四、系统实现及结果分析** | | | | | | | | | | 得分 | | |  |
| 要求：对如下内容展开描述。   1. 系统实现过程中遇到的问题 2. 文法二义性问题   实验指导书给出的文法是具有二义性的，主要体现在表达式符号的优先级和if else结构上，因此对这两部分进行改进。  此外，原文法不能确定while和if else语句的控制范围，因此对这两部分加上begin、end符号以确定控制范围。  改进后的文法已经在第二部分给出，此处不再赘述。   1. 空符号的处理问题   在计算FIRST集和规约时，要对有空（）符号的产生式特殊处理，这部分很容易产生bug，需要不断调试。   1. 错误处理问题   错误恢复包括恐慌模式错误恢复和短语层次错误恢复，本程序采用的是恐慌模式的错误恢复。   1. 输出该句法分析器的分析表   语法分析表较大，这里仅仅展示小部分，全部的内容在“<LR_Analysis_Table.txt>”文件中给出。     1. 针对一测试程序输出其句法分析结果   测试样例如下。   1. proc maxminavg; 2. **integer**[3] a; 3. **integer** max; 4. **integer** min; 5. **integer** sum; 6. **integer** i; 7. **real** avg; 9. a[1]=1; 10. a[2]=2; 11. a[3]=3; 12. sum=0; 13. max=a[1]; 14. min=a[1]; 15. i=1; 17. **while** **true** **do** 18. **begin** 19. a=a+1; 20. **end** 22. **if** a[i]>max **then** 23. **begin** 24. max=a[i]; 25. **end** 27. **if** a[i]>max **then** 28. **begin** 29. max=a[i]; 30. **end** 31. **else** 32. **begin** 33. a=1; 34. **end** 36. **while** **true** **do** 37. **begin** 39. **if** i<=3 **then** 40. **begin** 41. sum=sum+a[i]; 42. **end** 44. **if** a[i]>max **then** 45. **begin** 46. max=a[i]; 47. **end** 49. **if** a[i]<min **then** 50. **begin** 51. min=a[i]; 52. **end** 53. **else** 54. **begin** 55. call fuction(b,c,d) 56. **end** 58. i=i+1; 59. avg=sum\*1; 60. **end** 62. a[1]]>=1;; 64. **if** i<=<3 **then** 65. **begin** 66. a=a+1; 67. **end**   部分语法分析结果如下。全部结果在“<Productions.txt>”文件中给出。     1. 输出针对此测试程序对应的语法错误报告；   错误报告如下。       1. 对实验结果进行分析。   整体效果如下所示。    经过反复地测试与实验，从最后的实验结果来看，规约的产生式以及次序均正确，例如if else语句、while与if else的互相嵌套、call语句、record语句和多维数组等。  同时也对一些常见的错误进行了识别与恢复。例如最常见的“；”后又跟了一个“；”，系统自动忽略后一个分号；操作符不和规范，如出现“a>=>b”语句，系统对整个语句忽略掉。  结果表明，系统的结果与目标基本一致，系统目标达成。 | | | | | | | | | | | | | |
| 指导教师评语：  日期： | | | | | | | | | | | | | |