首先将文字描述的语法符号化

Program = SubProgram

SubProgram = begin DeclarationTable ; ExecuteTable end

DeclarationTable = DeclarationStatement | DeclarationTable ; DeclarationStatement

DeclarationStatement = VariableDeclaration | FunctionDeclaration

VariableDeclaration = integer Variable

Variable = Identifier

Identifier = Alphabet | Identifier Alphabet | Identifier Number

Alphabet = a | b | … | z

Number = 0 | 1 | 2 | … | 9

FunctionDeclaration = integer function Identifier ( Parameter ) ; FunctionBody

Parameter = Variable

FunctionBody = begin DeclarationTable ; ExecuteTable end

ExecuteTable = ExecuteStatement | ExecuteTable ; ExecuteStatement

ExecuteStatement = ReadStatement | WriteStatement | AssignmentStatement | ConditionStatement

ReadStatement = read ( Variable )

WriteStatement = write ( Variable )

AssignmentStatement = Variable := ArithmeticExpression

ArithmeticExpression = ArithmeticExpression-Item | Item

Item = Item \* Factor | Factor

Factor = Variable | Constant | FunctionCall

FunctionCall = Identifier ( Parameter )(这个PPT上没有，但是程序中确实是用到了的)

Constant = UnsignedInteger

UnsignedInteger = Number | UnsignedInteger Number

ConditionStatement = if ConditionExpression then Execute else Execute

ConditionExpression = ArithmeticExpression RelationOperator ArithmeticExpression

RelationOperator = < | <= | > | >= | = | <>

消除左递归

Program = SubProgram

SubProgram = begin DeclarationTable ; ExecuteTable end

DeclarationTable = DeclarationStatement DeclarationTable2

DeclarationTable2 = ; DeclarationStatement DeclarationTable2 | $

DeclarationStatement = VariableDeclaration | FunctionDeclaration

VariableDeclaration = integer Variable

Variable = Identifier

(下面这四个推倒式在识别标识符，而这一步在词法分析的过程中已经做过了其实)

Identifier = Alphabet Identifier2

Identifier2 = Alphabet Identifier2 | Number Identifier2 | $

Alphabet = a | b | … | z

Number = 0 | 1 | 2 | … | 9

FunctionDeclaration = integer function Identifier ( Parameter ) ; FunctionBody

Parameter = Variable

FunctionBody = begin DeclarationTable ; ExecuteTable end

ExecuteTable = ExecuteStatement ExecuteTable2

ExecuteTable2 =; ExecuteStatement ExecuteTable2 | $

ExecuteStatement = ReadStatement | WriteStatement | AssignmentStatement | ConditionStatement

ReadStatement = read ( Variable )

WriteStatement = write ( Variable )

AssignmentStatement = Variable := ArithmeticExpression

ArithmeticExpression = Item ArithmeticExpression2

ArithmeticExpression2 = -Item ArithmeticExpression2 | $

Item = Factor Item2

Item2 = \* Factor Item2 | $

Factor = Variable | Constant | FunctionCall

FunctionCall = Identifier ( Parameter )(这个PPT上没有，但是程序中确实是用到了的)

(注意下面三个常量的识别也是在词法分析的时候就完成了的)

Constant = UnsignedInteger

UnsignedInteger = Number UnsignedInteger2

UnsignedInteger2 = Number UnsignedInteger2 | $

ConditionStatement = if ConditionExpression then ExecuteStatement else ExecuteStatement

ConditionExpression = ArithmeticExpression RelationOperator ArithmeticExpression

RelationOperator = < | <= | > | >= | = | <>

另外有个很关键的地方是，无论是在子程序还是方法体的语法识别里面，如何区分申明语句和执行语句表的界限呢？我的方法是给生命语句表一个终结符$,也就是当声明语句表无法再识别声明语句的时候自动转到执行语句表