实验三 自顶向下的语法分析程序设计

**1.** 实验内容

任务一：运行**TINY**语言的语法分析程序**TINYParser**，理解**TINY**语言语法分析器的实现。

其中，TINY语言的词法与实验二相同，TINY语言的文法描述如下：

program -> stmt-seq

stmt-seq -> stmt-seq;stmt | stmt

stmt -> if-stmt|repeat-stmt|assign-stmt|read-stmt | write-stmt

if-stmt -> if exp then stmt-seq end | if exp then stmt-seq else stmt-seq end

repeat-stmt -> repeat stmt-seq until exp

assign-stmt -> id:= exp

read-stmt -> read id

write-stmt -> write exp

exp -> simp-exp cop simp-exp | simp-exp

cop -> < | =

simp-exp -> simp-exp addop term |term

term -> term mulop factor | factor

factor -> (exp) |num |id

addop -> + | -

mulop -> \* | /

对于如下TINY示例程序：

{A sample TINY program}

read x;

if 0<x then

fact:=1;

repeat

fact:=fact\*x;

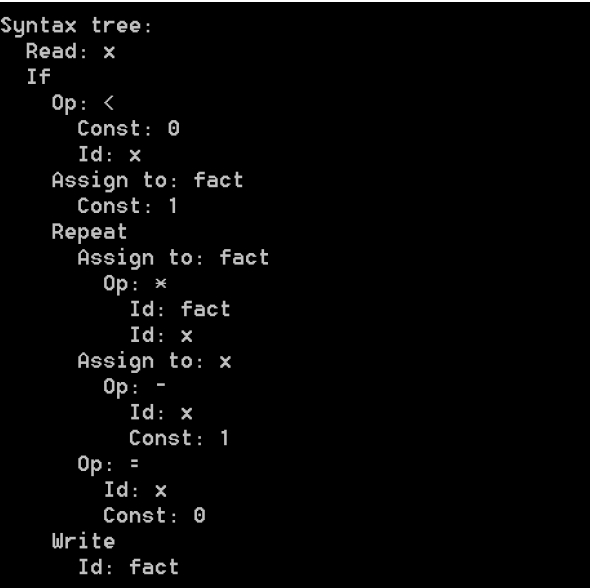
x:=x-1

until x=0;

write fact

end

TINY语法分析器的输出语法树为：



具体的语法树结构在TINY\_Syntax.pptx 里面描述，结合TINYParser 代码理解语法树构造。

任务一要求：根据**TINY** 语法，自己编写至少一个另外的**TINY** 测试程序， 运行**TINYParser** 语法分析器，观察程序运行流程，得到正确的运行结果。

任务二：基于TinyParser 语法分析器，实现拓展语言TINY+的语法分析器。

其中，TINY+语言的词法与实验二相同，TINY+语言的文法描述如下（注： 此处为了描述方便，对上下文无关文法的产生式表示进行了扩充，允许在产生式右部使用类似正则表达式的表示，例如第5 条产生式右部花括号{ , identifier }代表\*闭包。其中红色部分为TINY+文法更新的部分，其余部分为TINY 文法原有的产生式：

1 program -> declarations stmt-sequence

2 declarations -> decl ***;*** declarations |ε

3 decl -> type-specifier varlist

4 type-specifier -> ***int*** | ***bool*** | ***string***

5 varlist -> ***identifier { , identifier }***

6 stmt-sequence -> statement { ***;*** statement }

7 statement -> if-stmt | repeat-stmt | assign-stmt | read-stmt | write-stmt | while-stmt

8 while-stmt -> ***do*** stmt-sequence ***while*** bool-exp

9 if-stmt -> if exp then stmt-seq end | if exp then stmt-seq else stmt-seq end

10 repeat-stmt -> ***repeat*** stmt-sequence ***until*** exp

11 assign-stmt -> ***identifier:=***exp

12 read-stmt -> ***read identifier***

13 write-stmt -> ***write*** exp

14 exp -> simp-exp cop simp-exp | simp-exp

15 cop -> < | =

16 simp-exp -> simp-exp addop term |term

17 term -> term mulop factor | factor

18 factor -> (exp) |num |id

19 addop -> + | -

20 mulop -> \* | /

TINY+语言的文法主要添加了声明语句及while 语句，对于如下TINY+测试代码：

{this is an example}

int A,B;

bool C;

string D;

D:= 'scanner';

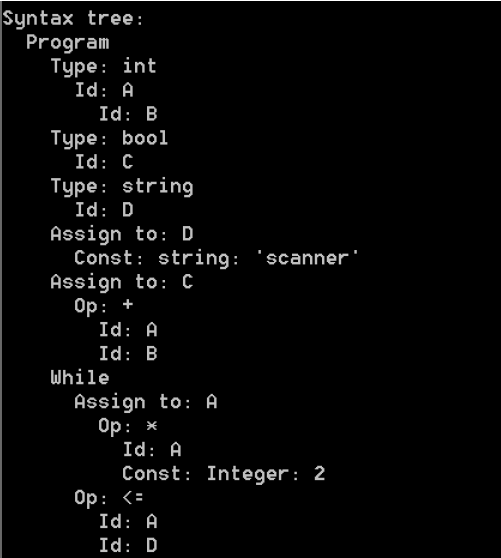
C:=A + B;

do

A:=A\*2

while A<=D

应得到以下TINY+语法分析结果（语法树）：



任务二要求：根据**TINY+**语法，修改给定的**TINY** 语法分析器，实现更新的**TINY+**语法分析器，成功实现对上述示例程序的语法分析。并根据**TINY+**文法的定义，编写至少一个另外的**TINY+**测试程序，对该测试程序完成语法分析，得到正确的语法分析结果。

**2.** 实验要求

完成任务一及任务二的要求；

 使用提供的模板撰写实验报告，不要增删，要求内容详实，包含课本上的理论描述，针对本实验的具体设计、思路描述，关键代码片段及其注释，及实验输入输出屏幕截图；

在截止日期前将代码、实验报告、测试文件（如有）等所有实验相关文件都放在一

个文件夹下面，并压缩到一个压缩包“学号\_姓名\_实验三.rar”上传至指定网站。如果提交了多次，请在文件后面加上“更新1，更新2”

周二晚上：http://xzc.cn/o5vwcYH7vV

周三下午：http://xzc.cn/ovBFS44jL8

周三晚上：http://xzc.cn/e4pvQ2hH4B

周四下午：http://xzc.cn/LhiSsgtGtH