

语法分析

0 概述

根据词法分析设计一个语法分析

1 实验设计

1.1 算法实现原理

自下而上的语法分析：

- 1) 从分析树的底部（叶节点）向顶部（根节点）方向构造分析树；
- 2) 可以看成是将输入串 w 归约为文法开始符号 S 的过程；
- 3) 自上而下的语法分析采用最左推导方式，自底向上的语法分析采用最左归约方式。

1.2 算法设计与分析

1.优先关系：

$a < b$ ：当且仅当文法 G 中有形如 $A \rightarrow \dots aR \dots$ 的产生式而 $R \Rightarrow +b \dots$ 或 $R \Rightarrow Qb \dots$ ， a 的优先级低于 b ；

$a = b$ ：当且仅当文法 G 中有形如 $A \rightarrow \dots ab \dots$ 或者 $A \rightarrow \dots aQb \dots$ 的产生式， a 的优先级等于 b ；

$a > b$ ：当且仅当文法 G 中有形如 $A \rightarrow \dots Rb \dots$ 的产生式，而 $R \Rightarrow + \dots a$ 或 $R \Rightarrow + \dots aR$ ， a 的优先级高于 b 。

如果一个算符文法 G 中的任何终结符对 (a, b) 至多满足下述三关系之一： $a < b, a = b, a > b$ ，则称 G 是一个算符优先文法。

2.制定优先级：

| | + | - | * | / | 标识符 | 无符号整数 | (|) | # |
|-------|---|---|---|---|-----|-------|---|---|---|
| + | = | = | < | < | < | < | < | > | > |
| - | = | = | < | < | < | < | < | > | > |
| * | > | > | = | = | < | < | < | > | > |
| / | > | > | = | = | < | < | < | > | > |
| 标识符 | > | > | > | > | | | | > | > |
| 无符号整数 | > | > | > | > | | | | > | > |
| (| < | < | < | < | < | < | < | = | > |
|) | > | > | > | > | | | = | > | > |
| # | < | < | < | < | < | < | < | < | = |

2 实验结果说明与分析

2.1 实验设置

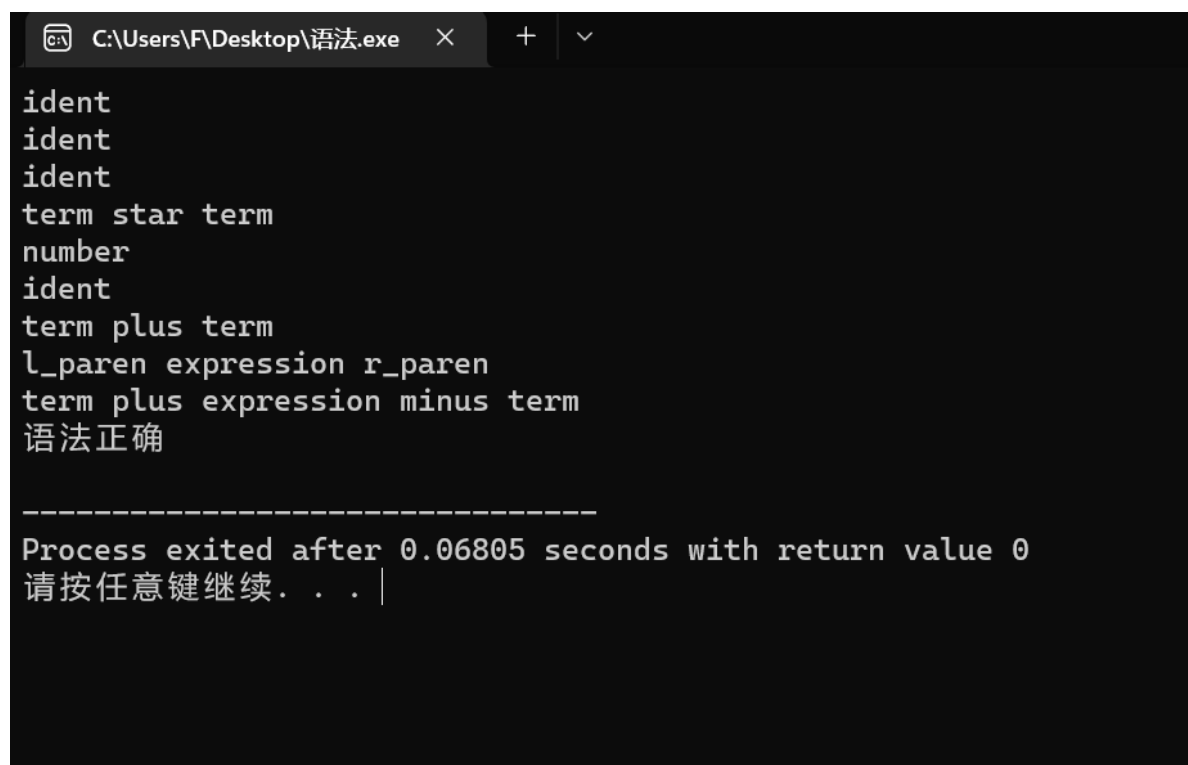
输入设计：

```
(ident,a)
(plus,+)
(ident,b)
(star,*)
(ident,c)
(minus,-)
(l_paren,(
(number,123456)
(plus,+)
(ident,b)
(r_paren,))
```

输出设计：

对于语法正确的表达式，报告“语法正确”；
对于语法错误的表达式，报告“语法错误”，指出错误原因

2.2 实验结果以及相应的分析



```
C:\Users\F\Desktop\语法.exe
ident
ident
ident
term star term
number
ident
term plus term
l_paren expression r_paren
term plus expression minus term
语法正确

-----
Process exited after 0.06805 seconds with return value 0
请按任意键继续. . .
```

3 总结

1、判断一个输入串是否符合语法规则：

从文法的起始符出发进行句子的推导，即自上而下的分析

从句子本身出发，进行归约，看能否把句子规约为到起始符，即自下而上的规约

2、基本思想：最左推导，本质上是一种试探过程。

3、构造方法：让每个非终结符号对应一个递归子程序。让候选式与输入串匹配，返回真值或者假。

