

# 字符串表达式解析算法

Ming

2008-10-30

字符串的扫描转化为单词链的过程比较简单，仅仅是按照数字字符和符号分割为单词，形成单词链，这里主要介绍一下将单词链转化为表达式树的步骤。表达式树的计算是一个简单的递归过程：

先判断左右子是否都空，是则看字符串是否空，是则返回数字，否则从变量表查找变量值返回；如果仅右子空，则执行单目运算，计算左子的递归结果的单目运算返回上一级；左右子都不空，则执行二目运算，计算左右子递归结果的二目运算结果返回上一级。

对于表达式单词链转化为树的过程需要两个堆栈，一个符号堆栈，另一个是数字堆栈。还有两个符号优先级判断函数----栈内优先级函数 ISLv()和栈外优先级函数 OSLv()。

栈外优先级		栈内优先级	
(	9	Func()	8
Func()	7	^	6
^	5	* /	4
* /	3	+ -	2
+ -	1	(	0

扫描的时候分两种模式，一种是数字和单目运算符模式，另一种是二目运算符模式。两种模式碰到不同类型单词会执行不同操作：

运算符类型	数字模式	二目运算符模式
数字	压入数字栈	抛出错误
二目运算符	抛出错误	该运算符栈外优先级低则 <b>出栈运算</b> 直到其优先级大于栈顶符号的栈内优先级，再将其压入。
字母	判断是否为函数，是函数则只要优先级不够就 <b>出栈运算</b> ，直到能压入再压入符号栈；否则直接压入数栈（变量）	抛出错误
左括号	压入符号栈	抛出错误
右括号	一直 <b>出栈运算</b> 直到栈顶为左括号，再将左括号出栈（右括号不如栈，仅用于和左括号匹配）	同左

出栈运算的方式如下：

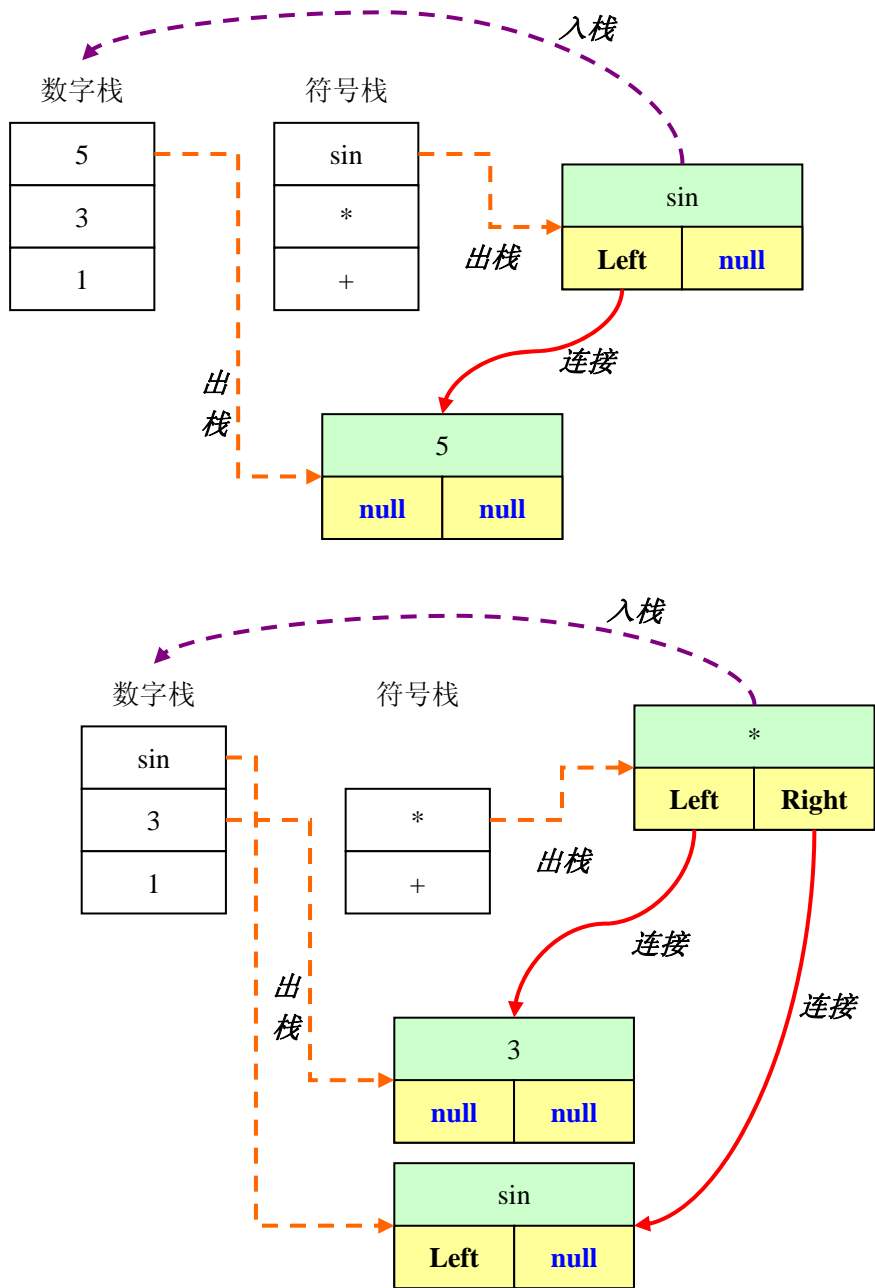
1.弹出栈顶运算符

2.判断运算符类型：

Case 单目运算：从数栈弹出一个数，连接到运算符 Left，扔回数栈

Case 二目运算：从数栈弹出两个数，连接到运算符 Left 和 Right，扔回数栈

出栈运算的示例见下图：



在表达式整个单词链扫描完成后，只要符号栈非空就进行 **出栈运算**。最后判断是否数字栈只有一个元素，是则把其返回。（即获得到计算树的根）