

实验 6 表达式求值

6.1 实验目的

- (1) 深入理解栈的特点及其描述方法。
- (2) 能够在两种存储结构上实现栈抽象数据类型实现。
- (3) 掌握栈的几个典型应用算法，能灵活应用栈解决实际问题。

6.2 预备知识

6.2.1 栈的类型定义（顺序存储结构）

```
#define MAXSIZE 100 //初始分配容量
typedef struct
{
    SElemType *base; //栈底指针
    SElemType *top; //栈顶指针
    int stacksize; //栈的最大容量
}SqStack; //栈类型定义
```

6.2.2 栈的几个最基本的操作

- (1) void InitStack(SqStack &S)
构造一个空栈 S。
- (2) void DestroyStack (SqStack &S)
释放栈 S 占用的内存空间。
- (3) int StackEmpty(SqStack S)
若 S 为空栈，则返回 1，否则返回 0。
- (4) void GetTop(Stack S, SElemType &e)
若栈不空，用 e 返回栈顶元素。
- (5) void Push(Stack &S, SElemType e)
将元素 e 入栈。
- (6) void Pop(Stack &S, SElemType &e)

若栈不空，删除栈顶元素，用 e 返回其值。

6.3 实验题目——表达式求值

利用栈实现算术表达式的求值，表达式中可包含加+、减（负）-、乘*、除/、乘方^、括号()运算符，操作数可以为浮点数。可采用直接求中缀表达式的方法，也可采用先转换成后缀表达式后再求值的方法（参看课件）。

实现时需注意如下：

（1）带小数点的数值生成（理解整数数值的生成，小数部分的处理与之类似）。

（2）考虑负号的情况。负号与减号形式上一样，如何区分？

- 输入的表达式串中第 1 个字符是 '-'
 - 可断定此 '-' 是负号
- 其余的 '-'
 - 若紧接在 '(' 之后的 '-' 可断定是负号

（3）负号的处理

- **方法 1：**若已断定是负号，可先压入数值 0 进入操作数栈，如此，可将负号看作减号。
- **方法 2：**若已断定是负号，则紧接其后的数字字符转换成数值后，要将其对应的相反数(负数)压入操作数栈。

6.4 实验要求

- （1）栈的类型定义和基本操作函数声明放在 Sqstack.h 文件；
- （2）将其他函数的声明放在 other.h 文件；
- （3）基本操作函数的实现放在 Sqstack.cpp 文件；
- （4）将其他函数的实现放在 other.cpp 文件；
- （5）测试程序放在 ExpressionTestApp.cpp 中。