

3、栈和队列

3.1 栈

1、抽象数据类型栈的定义

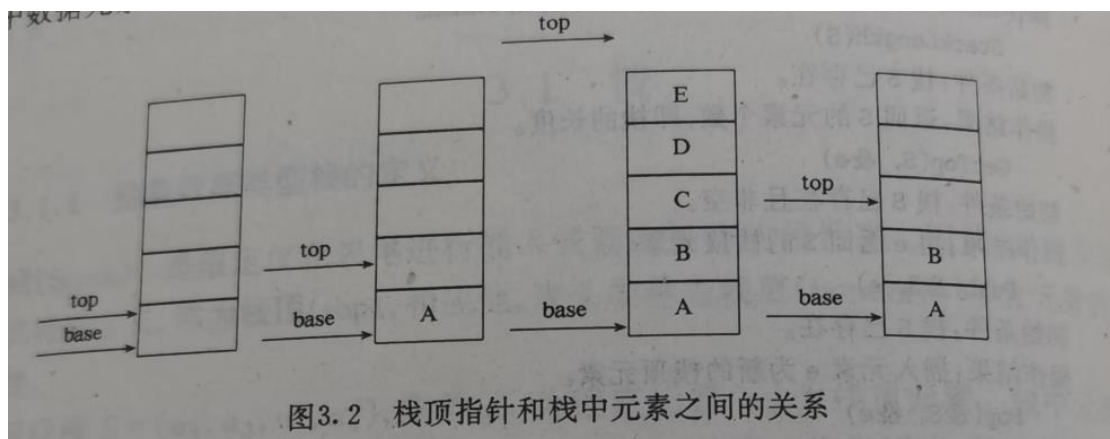
栈式限定仅在表尾进行插入或删除操作的线性表。因此,对栈来说**表尾端**有其特殊含义,称为**栈顶**。相应地, **表头端**称为**栈底**。

2、栈的表示和实现

①顺序栈:利用一组地址连续的存储单元依次存放自栈底到栈顶的数据元素,同时附设 top 指针指示栈顶元素再顺序栈中的位置。通常,以 $top=0$ 表示空栈。(注意:若如此做,由于栈的大小未知,因此设置一个最大数(例如 100),若超过了这个数,则再次申请 100 个)

栈的数据结构定义:

```
typedef struct
{
    SElemType *base;
    SElemType *top;
    int stacksize;
}SqStack;
```



进栈出栈如图所示

进则 $top++$, 出则 $top--$

3.2 栈的应用举例

- 1、数制转换 (√)
- 2、括号匹配的检验 (√)
- 3、行编辑程序
- 4、表达式求值

3.3 栈的递归实现

通常：当在一个函数的运行期间调用另一个函数时，在运行被调用函数之前，系统需先完成三件事：①将所有的实在参数、返回地址等信息传递给被调用函数保存②为被调用函数的局部变量分配存储区③将控制转移到被调用函数的入口。

而从被调用函数返回调用函数之前，系统也要完成三样工作：①保存被调用函数的计算结果②释放被调用函数的数据区③依照被调用函数保存的返回地址将控制转移到调用函数。

当有多个函数嵌套调用时，按照“后调用先返回”的原则，上述函数之间的信息传递和控制转移必须通过栈来实现。

3.4 队列

1、 队列：是一种先进先出（First In First Out, FIFO）的线性表（与栈相反）。它只允许在表的一端进行插入，而在另一端进行删除。

2、 链队列——队列的链式表示和实现

即前一个元素指向后一个元素，队头有一个指针，队尾还有一个指针。

（与链表相似，详见书 P61）

3.5 练习实例：离散事件模拟