

第三讲 堆栈stack（特殊的线性结构）

回顾上节课所学知识点

- 许福林：空表，哨兵位的作用是什么呢？

<https://www.zhihu.com/question/27155932>

- 双向链表的插入与删除；

1. 什么是堆栈 stack

- 计算机如何进行表达式求值？

[例] 算术表达式 $5+6/2-3*4$ 。正确理解：

$$5+6/2-3*4 = 5+3-3*4 = 8-3*4 = 8-12 = -4$$

□ 由两类对象构成的：

➤ 运算数，如2、3、4

➤ 运算符号，如+、-、*、/

□ 不同运算符号优先级不一样

• 为什么需要堆栈这种数据结构？

后缀表达式

➤ 中缀表达式：运算符号位于两个运算数之间。如， $a + b * c - d / e$

➤ 后缀表达式：运算符号位于两个运算数之后。如， $a b c * + d e / -$

【例】 $6\ 2\ /\ 3\ -\ 4\ 2\ *\ +\ =\ ?$

后缀表达式求值策略：从左向右“扫描”，逐个处理运算数和运算符号

1. 遇到运算数怎么办？如何“记住”目前还未参与运算的数？
2. 遇到运算符号怎么办？对应的运算数是什么？

启示：需要有存储方法，能顺序存储运算数，并在需要时“倒序”输出！

• 【例】 $6\ 2\ /\ 3\ -\ 4\ 2\ *\ +\ =\ ?$

对象: 6 (运算数)	对象: 2 (运算数)
对象: / (运算符)	对象: 3 (运算数)
对象: - (运算符)	对象: 4 (运算数)
对象: 2 (运算数)	对象: * (运算符)
对象: + (运算符)	Pop: 8

● 堆栈的抽象数据类型描述

堆栈 (Stack)：具有一定操作约束的线性表
 ➤ 只在一端（栈顶，Top）做 **插入、删除**

类型名称: 堆栈 (Stack)

数据对象集: 一个有0个或多个元素的有穷线性表。

操作集: 长度为MaxSize的堆栈 $S \in \text{Stack}$, 堆栈元素 $\text{item} \in \text{ElementType}$

- 1、**Stack CreateStack(int MaxSize)**: 生成空堆栈，其最大长度为MaxSize;
- 2、**int IsFull(Stack S, int MaxSize)**: 判断堆栈S是否已满;
- 3、 **void Push(Stack S, ElementType item)**: 将元素item压入堆栈;
- 4、**int IsEmpty (Stack S)**: 判断堆栈S是否为空;
- 5、 **ElementType Pop(Stack S)**: 删除并返回栈顶元素;

● 堆栈的操作集合

(1) 堆栈的数据结构

栈的顺序存储结构通常由一个一维数组和一个记录栈顶元素位置的变量组成。

a是指向数组的指针；top表示栈顶；capacity表示数组的实际容量。

(我们这里实现的是一个动态的数组栈，空间不够时，可以动态增容)

(2) 栈的初始化

初始化：

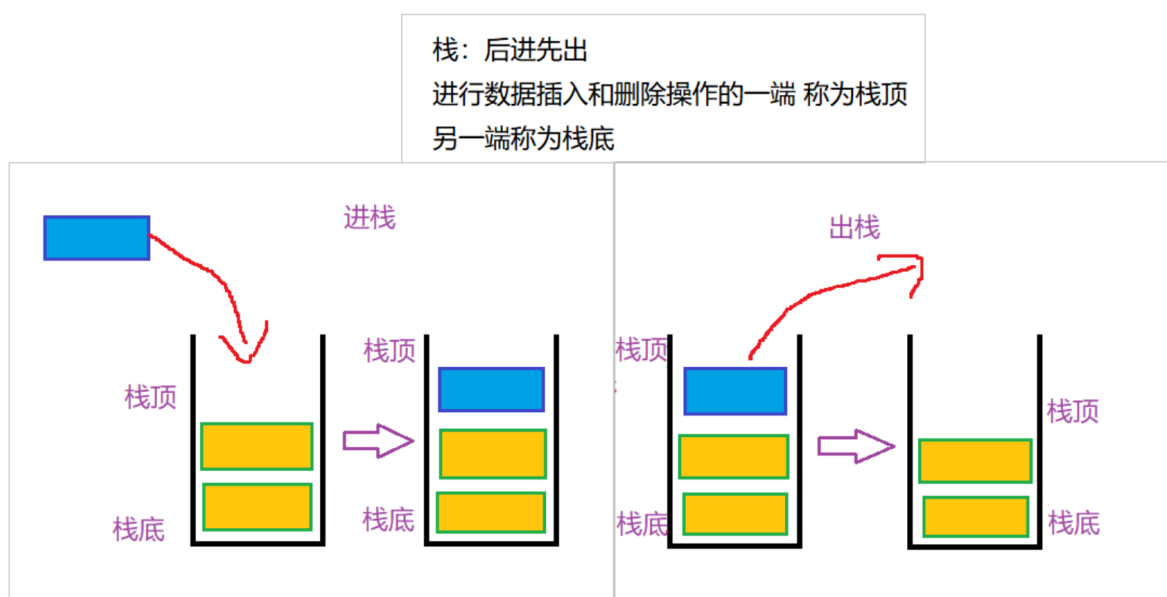
- 这里把top初始化成 0，则top就代表栈顶的下一个数据的下标位置。
- (这个时候的top也代表了栈内数据的个数) 如果这里把top初始化成 -1，则top就代表栈顶数据的下标位置。

(3) 栈顶插入数据(入栈)

压栈：栈的插入操作叫做进栈 / 压栈 / 入栈，入数据在栈顶。

对于数组进行插入操作时，需要注意的是：

- 要首先检查数组空间够不够，不够了就要进行扩容操作；空间够的话，就可以进行
- 插入操作了
 - 数组存数；
 - top栈顶变化



(4) 删除栈顶数据(出栈)

出栈：栈的删除操作叫做出栈。出数据也在栈顶。

对于删除操作，需要注意的是：

如果栈已经为空（ $top==0$ ）的时候，就不能进行删除操作了。

如果栈不为空（ $top!=0$ ）时，**让 $top--$** ，就完成了数组栈的删除操作了。

（5）删除栈顶数据(出栈)

首先栈不为空时，才有栈顶数据，所以要先判断栈是否为空。

栈不为空时，取出栈顶数据（ $ps \rightarrow a[ps \rightarrow top - 1]$ ）就可以了。

（6）栈内的元素个数

top 的值就是栈内的元素个数了。

（7）判断栈的是否为空

判断栈是否为空：如果 $top==0$ ，则栈为空；如果 $top!=0$ ，则栈不为空。

（8）栈的销毁

释放a指针所指向的连续的空间，把a指针置空，把top和capacity置0.

选择题1:

1. 若进栈序列为 1,2,3,4 ，进栈过程中可以出栈，则下列不可能的一个出栈序列是 ()
- A 1,4,3,2
 - B 2,3,4,1
 - C 3,1,4,2
 - D 3,4,2,1