




数据结构
(C语言版) (第2版)
栈和队列
栈和队列基本概念

主讲教师：汪红松





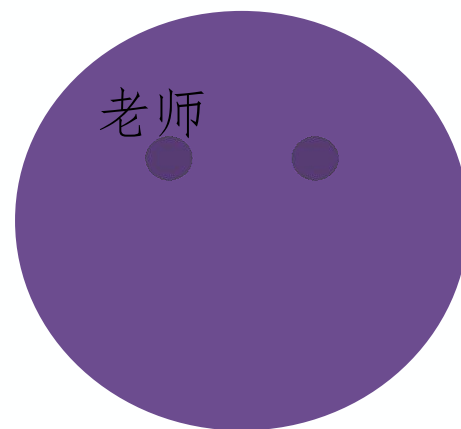
教学目标

- 01
OPTION 熟悉栈的类型定义，栈的**顺序**和链式存储的表示和实现；
 - 02
OPTION 了解栈与递归的实现，递归程序转换非递归程序的方法；
 - 03
OPTION 熟悉队列的类型定义，队列的顺序和链式存储的表示和实现；
 - 04
OPTION 能够运用栈和队列的特点，正确选用栈和队列解决相应的应用问题，如迷宫求解、表达式求值、杨辉三角形、模拟汽车加油站等问题。
- 

教学内容 Contents

- 一 栈和队列基本概念
- 二 栈的表示和操作
- 三 栈与递归
- 四 队列的表示和操作

- 一、栈
- 二、队列
- 三、栈、队列与一般线性表的区别





栈 (Stack)

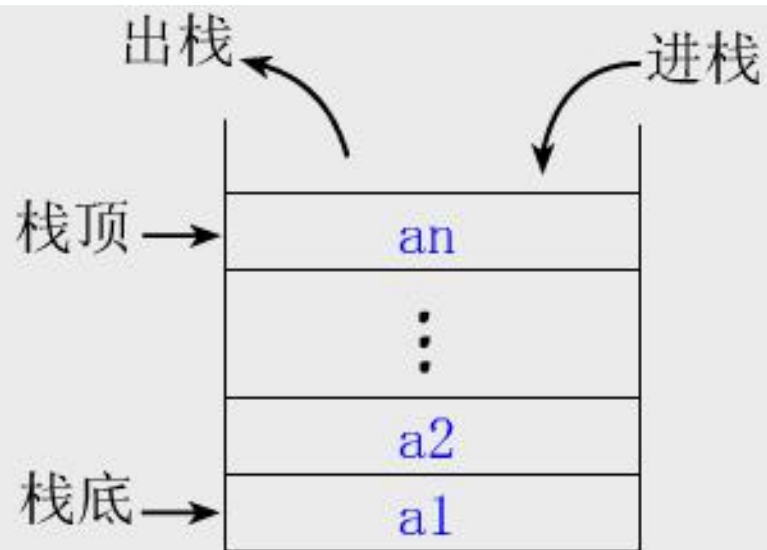
1. 定义
2. 逻辑结构
3. 存储结构
4. 运算规则
5. 实现方式



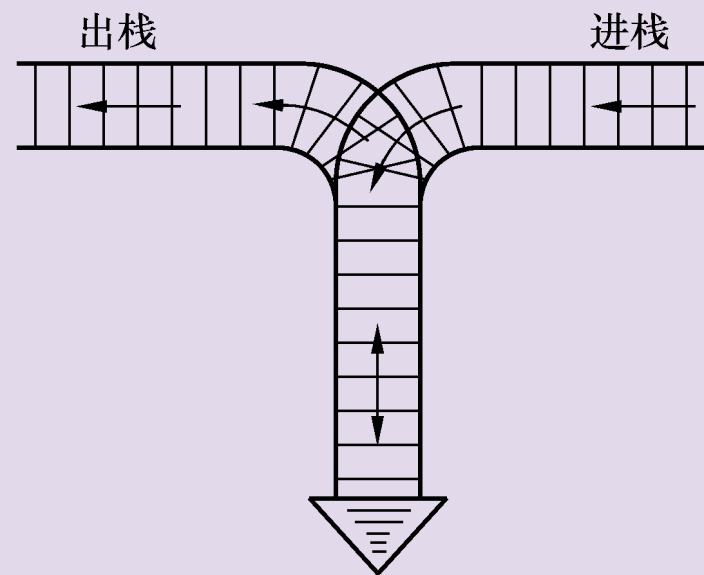
队列 (Queue)

1. 定义
2. 逻辑结构
3. 存储结构
4. 运算规则
5. 实现方式

一、栈



栈示意图



用铁路调度站表示栈

▶▶▶ 一、栈

01 定义

OPTION

只能在表的一端（栈顶）进行插入和删除运算的线性表。

02 逻辑结构

OPTION

与线性表相同，仍为一对一关系。

03 存储结构

OPTION

用顺序栈或链栈存储均可，但以顺序栈更常见。

04 运算规则

OPTION

只能在栈顶运算，且访问结点时依照后进先出（LIFO）或先进后出（FILO）的原则。

05 实现方式

OPTION

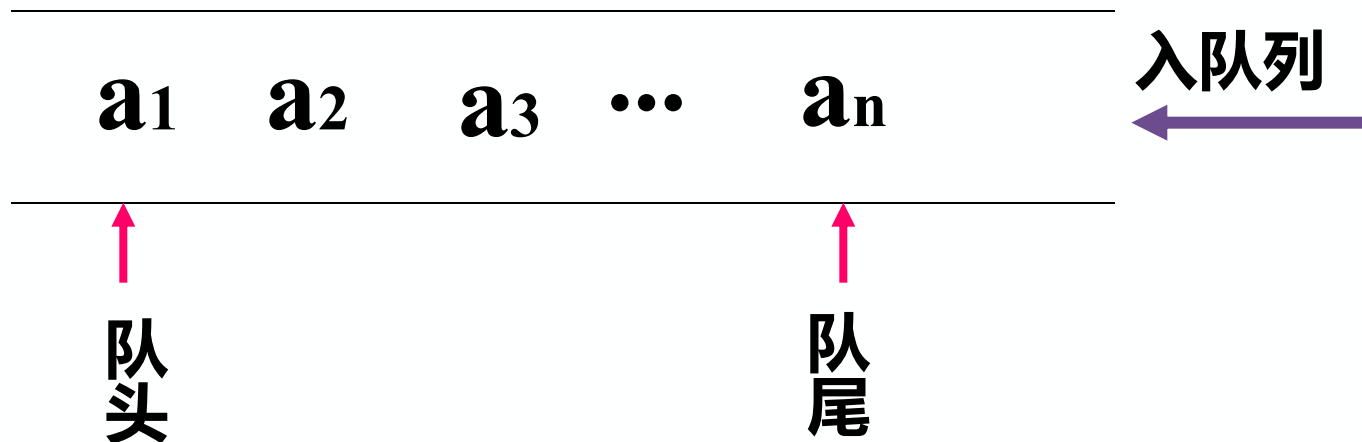
关键是编写入栈和出栈函数，具体实现依顺序栈或链栈的不同而不同。
基本操作有入栈、出栈、读栈顶元素值、建栈、判断栈满、栈空等。

二、队列

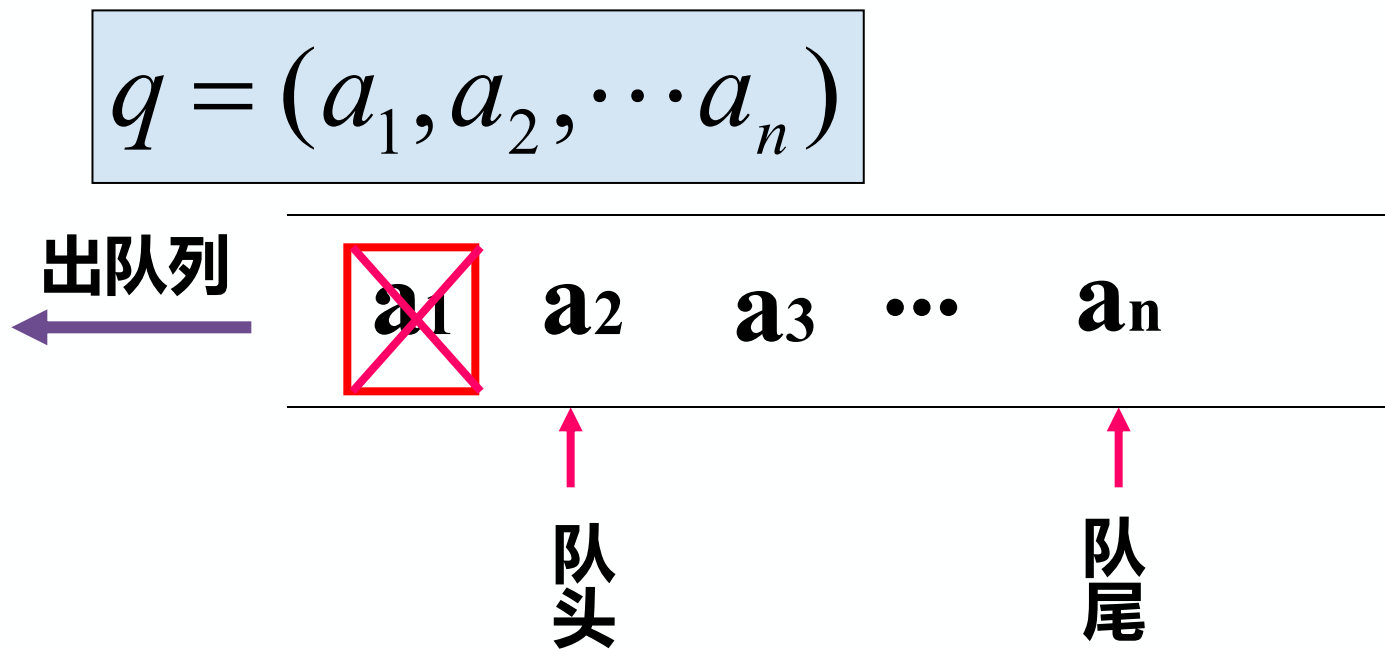
队列是一种先进先出 (FIFO) 的线性表. 在表一端插入, 在另一端删除。



$$q = (a_1, a_2, \dots, a_n)$$



二、队列



二、队列

$$q = (a_1, a_2, \cdots a_n)$$

出队列
←

~~a_1~~

~~a_2~~

a_3

\cdots

a_n

↑
队头

↑
队尾

▶▶▶ 二、队列

01 定义

OPTION

只能在表的一端（**队尾**）进行插入，在另一端（**队头**）进行删除运算的**线性表**。

02 逻辑结构

OPTION

与线性表相同，仍为**一对一**关系。

03 存储结构

OPTION

用**顺序队列**或**链队**存储均可。

04 运算规则

OPTION

先进先出（FIFO）。

05 实现方式

OPTION

关键是编写**入队**和**出队**函数，具体实现依顺序队或链队的不同而不同。

▶▶▶ 三、栈、队列与一般线性表的区别

栈、队列是一种特殊（**操作受限**）的线性表。

区别：仅在于**运算规则**不同。

一般线性表

逻辑结构：一对一

存储结构：顺序表、链表

运算规则：**随机、顺序存取**

栈

逻辑结构：一对一

存储结构：顺序栈、链栈

运算规则：**后进先出**

队列

逻辑结构：一对一

存储结构：顺序队、链队

运算规则：**先进先出**

1. 栈的定义和特点
2. 队列的定义和特点
3. 栈和队列与一般线性表的区别