

线性表

1. 线性表可以是空表，当元素个数为0时，即为空表
2. 头元素，尾元素，**直接前驱，前驱，直接后继，后继**，

顺序表

1. 顺序表定位每一个地址所需的时间是相同的，这表明顺序表具有按数据元素的序号随机存取的特点，因此又称作**线性表的随机存储结构**
2. 顺序表的基本操作

种类	按位置查找	按值查找	插入操作	删除操作
复杂度	$O(1)$	$O(n)$, $ASL = \frac{n+1}{2}$	$O(n)$	$O(n)$

除此之外，还有创建，**销毁，清空**，判断是否为空，求前驱，求后继

3. 优点
 1. 方法简单，容易实现
 2. 不用为表示节点之间的逻辑关系而增加额外的存储开销
 3. 可按序号随机存取顺序表中的元素
4. 缺点
 1. 插入删除操作时，涉及大量元素的移动，时间效率低
 2. 需要预先分配存储空间，容易造成空间的不够用或者空间的浪费

单链表

1. 带头结点的单链表
 1. 操作及时间复杂度

操作	按值查找	按位置查找	添加	删除
复杂度	$O(n)$	$O(n)$	$O(n)$	$O(n)$

2. 线性表的创建过程有两种，从头结点处插入，和从尾结点处插入，由于从尾结点处插入需要跟踪尾结点的位置，因此，我们通常使用第一种方式，但是在**插入时要求数据元素以相反的顺序读入**
2. 不带头结点的单链表
 1. 注意由于不带头结点的链表第一个节点的处理方法和其他节点的不同，因此，在许多情况下需要**对第一个节点进行特殊处理**
3. 带头结点的好处
 1. 链表的第一个节点和其他节点的处理方法一样，不用特殊处理
 2. 无论链表是否为空，因为头指针的存在，使得空表与非空表的处理一样
4. 其他特殊的单链表
带尾指针的，双向的，循环的等等

栈

1. 相关名词

1. 栈顶，栈底，出栈，入栈，空栈（**当没有元素时成为空栈**）

2. 先进后出，后进先出

3. 顺序栈

1. 顺序存储的栈叫顺序栈，对于顺序栈，由于栈底的位置是不变的，**因此我们将数组的首部作为栈底**
2. 顺序栈的存储空间的分配有两类，静态分配（提前分配一块连续的地址），动态分配（使用 malloc）

4. 优缺点

1. 优点：由于栈底插入和删除操作均在栈顶，因此使用顺序结构存储栈在插入删除的过程中不涉及大量元素的移动，效率很高**。
2. 缺点：缺点在与难以预先分配合适大小的存储空间，可能会造成大量空间的浪费和空间的不够用的问题

5. 链栈

1. 由于插入和删除等操作在栈的一侧，而线性表的插入在头结点处比较方便，因此，我们将**单链表的首端作为栈顶，同时为了避免对空和非空的特殊处理，我们使用带头结点的线性表**

队列

1. 队头，队尾，出队，入队，

2. 先进先出，后进后出

3. 队列需要有两个指针front，和rear

4. 顺序队列

1. 我们的教材约定，队头指针指向队头元素的前一个位置，队尾指针指向队尾元素
2. 为了解决假溢出的问题，我们可以使用循环队列
3. 循环队列的空与非空的解决方法
 1. 使用计数器
 2. 单独表示
 3. 少使用一个元素

5. 链式队列

1. 封装的过程中，跟队列（栈）相关的东西封装在同一个结构体内部，注意不要和节点的结构体整混淆了
- 2.