# 1、将一维数组的元素逆序存储

方法：

a[0] a[n] 互换

a[1] a[n-1] 互换

循环次数为n/2，具体情况按照奇数偶数再看。

# 2、单链表

单链表是一种链式存取的数据结构，用一组地址任意的[存储单元](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E5%8D%95%E5%85%83)存放线性表中的[数据元素](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%85%83%E7%B4%A0)。链表中的数据是以结点来表示的，每个结点的构成：元素([数据元素](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%85%83%E7%B4%A0)的映象) + [指针](https://baike.baidu.com/item/%E6%8C%87%E9%92%88)(指示后继元素存储位置)，元素就是存储数据的存储单元，指针就是连接每个结点的地址数据。

单链表长度的探寻：指针一直往下找，直到为NULL，在这个过程中长度一直++。

单链表删除结点：

单链表增加结点：

单链表排序：

单链表逆置：

给出一个单链表，不知道结点N的值，怎么只遍历一次就可以求出中间结点。

# 3、双链表：

每个数据节点中都有两个指针，分别指向直接后继和直接前驱。所以从双向链表中的任意一个结点开始，都可以很方便访问其前后结点。

1. 插入：插入就是把前后指针都做好，要先判断是否是插入在最后一个位置，如果是，就要注意了，这个被插入的元素的后继指针要指向NULL。
2. 删除：如果删除的是最后一个结点，要把前一个结点的后继指针指向NULL。

# 4、循环链表：

每一个结点都可以作为头结点，为节点的后继指针指向头结点。

# 5、队列

先进先出。插入结点是从后面插入，删除结点或者说释放节点是从队列的头部删除。

# 6、栈

先进后出。

栈区由操作系统自动分配释放，存放函数的参数值，局部变量的值等。

但是在java中都是用来在RAM存储数据的地方，程序员不能直接地设置栈或堆。

栈空间有限

# 7、堆

先进先出。

堆区由程序员分配释放，若程序员不释放，在程序结束时由操作系统回收。

堆有很大的自由存储区

# 8、树、图、哈希表

拓扑排序：

1. 在有向图中选一个没有前驱的顶点并且输出
2. 从图中删除该顶点和所有以它为尾的弧（白话就是：删除所有和它有关的边）
3. 重复上述两步，直至所有顶点输出，或者当前图中不存在无前驱的顶点为止，后者代表我们的有向图是有环的，因此，也可以通过拓扑排序来判断一个图是否有环。

哈希表：根据关键码值而直接进行访问的数据结构，把关键码值映射到表中一个位置来访问记录，以加快查找的速度。这个映射函数叫做散列函数，存放记录的数组叫做散列表。

拉链发。

# 9、排序

# 10、时间复杂度