一、顺序表：

优点：

1. 顺序表的内存空间连续。
2. 尾插、尾删效率较高，时间复杂度是O(1)。
3. 支持随机访问，可以高效的按下标进行操作，时间复杂度是O(1)。
4. CPU高速缓存利用率更高

缺点：

1. 在顺序表中间插入或删除元素时都涉及到元素的移动，效率较低，时间复杂度为O(N)。
2. 顺序表长度固定，有时需要扩容。为了避免频繁增容，一般我们都按倍数去赠，用不完可能存在一定的空间浪费。

二、链表：

优点

1. 链表的内存空间不连续。
2. 如果知道要处理节点的前一个位置，则进行插入和删除的复杂度为O(1);
3. 如果不知道要处理节点的前一个位置，则进行插入和删除的复杂度为O(N)。
4. 头插、头删的效率高，时间复杂度是O(1)。
5. 没有空间限制，不会溢出，可以存储很多元素。

缺点：

1. 链表不支持随机访问（用下标访问），意味着，一些排序、二分查找等在这种结构上不适用。
2. 查找元素效率低，需要遍历节点，时间复杂度是O(n)。
3. 链表存储一个值，同时要存储链接指针，也有一定的消耗。
4. CPU高速缓存利用率更高

三、总结

- 当线性表的长度变化不大、易于确定其大小时，采用顺序表作为存储结构。
- 若线性表主要操作是查找。很少进行插入或删除操作时，采用顺序表作为存储结构。
- 对于频繁进行插入和删除的线性表，则应该使用链表作为存储结构。