数据结构1 线性结构

数据结构是指数据元素的结合及元素间的相互关系和构造方法。元素之间的相互关系是数据的逻辑结构，元素关系的存储形式成为存储结构。数据结构按照逻辑关系的不同分为线性结构和非线性结构两大类。其中线性结构是最基本的结构，元素顺序排列，常见的有线性表、栈、队列、数组、串。

一、线性表

1.线性表是最简单也是最常用的一种线性结构。一个线性表示n(n>=0)个元素的有限序列，非空线性表的特点为：

存在唯一的一个“第一个”元素；

存在唯一的一个“最后一个”元素；

除第一个元素外，序列中的每个元素均只有一个直接前驱；

除最后一个元素外，序列中的每个元素均只有一个直接后驱；

2.关于线性表的存储结构，有顺序存储和链式存储两种方式。顺序存储是用一组地址连续的存储单元依次存储线性表中的数据元素，从而使得逻辑上相邻的两个元素在物理位置上也相邻。链式存储则使用结点来存储数据元素，结点分为数据域和指针域，数据域用于存储元素的值，指针域则存储元素的直接前驱或直接后驱的地址，结点之间通过指针域构成一个链表。若结点中只有一个指针域，则称为单链表。

顺序存储结构无须占用额外的空间来表示元素之间的逻辑关系，而且可以随机存取表中的元素，但插入、删除数据的操作比较麻烦，需要顺次移动很多元素；链表中各数据元素的结点地址则不要求是连续的，因此必须同时存储元素之间的逻辑关系，链表的插入、删除操作都比较方便，只需修改指针域的指向即可。

3.根据结点中指针域的设置方式，还有其他几种链表结构：

双向链表：每个阶段包含两个指针，分别指向直接前驱和直接后驱，这样可以从表中任意的结点出发，从两个方向遍历链表。

循环链表，在单向链表或双向链表的基础上，令表尾结点的指针指向第一个元素，这样就可以从任意结点开始遍历整个链表了。

1. 栈和队列

栈和队列的逻辑结构和线性表相同，但对其的运算有所限制。

1.栈

栈只能通过访问它的一端来实现数据的存储和检索，即对栈的操作时按照先进后出的原则进行的（Last In first out, LIFO），这一端称为栈顶，则另一端为栈底。不含任何元素的栈称为空栈。

栈的存储结构也分顺序和链式，顺序存储是用一组地址连续的存储单元依次存储自栈顶到栈底的数据元素，同时用指针top来指示栈顶元素的位置，这种结构的栈也成为顺序栈。使用这种存储方式需要预先定义栈的存储空间，栈的空间有限，所以元素入栈前，需要判断栈是否已满，否则会发生溢出。

链式存储的栈（链栈）没有溢出的问题，链栈也无须另外设置top指针

栈在表达式求值、计算机语言的实现、将递归过程转变为非递归过程的处理中有重要作用。

2.队列

队列则是一种先进先出(First in first out, FIFO)的线性表，但只能在队列的队尾(rear)插入元素，然后在队头(front)删除元素。

与顺序栈类似，顺序队列采用顺序存储结构，并需要设置队头指针和队尾指针。

链队列则为了便于操作，会添加一个头结点，并令头指针指向头结点，因此如果头指针和尾指针的值相同，则可判定队列为空。

队列常用于处理需要排队的场合，如打印队列、消息队列等。