

1.试比较顺序存储结构和链式存储结构的优缺点。在什么情况下用顺序表比链表好？

答：① 顺序存储时，相邻数据元素的存放地址也相邻（逻辑与物理统一）；要求内存中可用存储单元的地址必须是连续的。

优点：存储密度大（=1），存储空间利用率高。缺点：插入或删除元素时不方便。

②链式存储时，相邻数据元素可随意存放，但所占存储空间分两部分，一部分存放结点值， 另一部分存放表示结点间关系的指针

优点：插入或删除元素时很方便，使用灵活。缺点：存储密度小（<1），存储空间利用率低。顺序表适宜于做查找这样的静态操作；链表宜于做插入、删除这样的动态操作。

若线性表的长度变化不大，且其主要操作是查找，则采用顺序表；

若线性表的长度变化较大，且其主要操作是插入、删除操作，则采用链表。

	顺序表	链表
优点	1、方法简单，各种高级语言中都有数组，容易实现； 2、不用为表示结点间的逻辑关系而增加额外的存储开销，存储密度大； 3、具有按元素序号随机访问的特点，查找速度快。	1、插入、删除时，只要找到对应前驱结点，修改指针即可，无需移动元素； 2、采用动态存储分配，不会造成内存浪费和溢出。
缺点	1、插入删除操作时，需要移动元素，平均移动大约表中一半的元素，对元素较多的顺序表效率低。 2、采用静态空间分配，需要预先分配足够大的存储空间，会造成内存的浪费和溢出。	1、在有些语言中，不支持指针，不容易实现； 2、需要用额外空间存储线性表的关系，存储密度小 3、不能随机访问，查找时要从头指针开始遍历。

2.一棵度为2 的有序树与一棵二叉树有何区别？

答：一棵度为二的有序树与一棵二叉树的区别在于:有序树的结点次序是相对于另一结点而言的，如果有序树中的子树只有一个孩子时，这个孩子结点就无须区分其左右次序。

而二叉树无论其孩子数是否为2，均需确定其左右次序，也就是说二叉树的结点次序不是相对于另一结点而言而是确定的。

3.简述逻辑结构与存储结构的关系

答：数据的逻辑结构反映数据元素之间的逻辑关系（即数据元素之间的关联方式或“邻接关系”），数据的存储结构是数据结构在计算机中的表示，包括数据元素的表示及其关系的表示。

3.简述下列术语：数据，数据元素、数据对象、数据结构、存储结构、数据类型和抽象数据类型。

答：数据是对客观事物的符号表示。在计算机科学中是指所有能输入到计算机中并被计算机程序处理的符号的总称。

数据元素是数据的基本单位，在计算机程序中通常作为一个整体进行考虑和处理。数据对象是性质相同的数据元素的集合，是数据的一个子集。

数据结构是相互之间存在一种或多种特定关系的数据元素的集合。存储结构是数据结构在计算机中的表示。

数据类型是一个值的集合和定义在这个值集上的一组操作的总称。

抽象数据类型是指一个数学模型以及定义在该模型上的一组操作，是对一般数据类型的扩展。

4.试描述数据结构和抽象数据类型的概念与程序设计语言中数据类型概念的区别。

答：抽象数据类型包含一般数据类型的概念，但含义比一般数据类型更广、更抽象。一般数据类型由具体语言系统内部定义，直接提供给编程者定义用户数据，因此称它们为预定义数据类型。抽象数据类型通常由编程者定义，包括定义它所使用的数据和在这些数据上所进行的操作。在定义抽象数据类型中的数据部分和操作部分时，要求只定义到数据的逻辑结构和操作说明， 不考虑数据的存储结构和操作的具体实现， 这样抽象层

次更高，更能为其他用户提供良好的使用接口。

5.描述以下三个概念的区别：头指针，头结点，首元结点（第一个元素结点）。

头指针是指向链表中第一个结点的指针。

首元结点是指链表中存储第一个数据元素的结点。

头结点是在首元结点之前附设的一个结点，该结点不存储数据元素，其指针域指向首元结点，其作用主要是为了方便对链表的操作。它可以对空表、非空表以及首元结点的操作进行统一处理。

6.线性表的两种存储结构各有哪些优缺点？

答：线性表具有两种存储结构即顺序存储结构和链接存储结构。线性表的顺序存储结构可以直接存取数据元素，方便灵活、效率高，但插入、删除操作时将会引起元素的大量移动，因而降低效率；而在链接存储结构中内存采用动态分配，利用率高，但需增设指示结点之间关系的指针域，存取数据元素不如顺序存储方便，但结点的插入、删除操作较简单。

7.对于线性表的两种存储结构，如果有n个线性表同时并存，而且在处理过程中各表的长度会动态发生变化，线性表的总数也会自动改变，在此情况下，应选用哪一种存储结构？为什么？

答：应选用链接存储结构，因为链式存储结构是用一组任意的存储单元依次存储线性表中的各元素，这里存储单元可以是连续的，也可以是不连续的：这种存储结构对于元素的删除或插入运算是需要移动元素的，只需修改指针即可，所以很容易实现表的容量的扩充。

8.对于线性表的两种存储结构，若线性表的总数基本稳定，且很少进行插入和删除操作，但要求以最快的速度存取线性表中的元素，应选用何种存储结构？试说明理由。

答：应选用顺序存储结构，因为每个数据元素的存储位置和线性表的起始位置相差一个和数据元素在线性表中的序号成正比的常数。因此，只要确定了其起始位置，线性表中的任一个数据元素都可随机存取，因此，线性表的顺序存储结构是一种随机存取的存储结构，而链表则是一种顺序存取的存储结构。

9.在单循环链表中设置尾指针比设置头指针好吗？为什么？

答：设尾指针比设头指针好。尾指针是指向终端结点的指针，用它来表示单循环链表可以使得查找链表的开始结点和终端结点都很方便，设一带头结点的单循环链表，其尾指针为rear，则开始结点和终端结点的位置分别是 rear->next->next 和 rear，查找时间都是 O(1)。若用头指针来表示该链表，则查找终端结点的时间为O(n)。

10.什么是队列的上溢现象？一般有几几种解决方法，试简述之。

答：在队列的顺序存储结构中，设队头指针为front，队尾指针为rear，队列的容量（即存储的空间大小）为maxnum。当有元素要加入队列（即入队）时，若rear=maxnum，则会发生队列的上溢现象，此时就不能将该元素加入队列。对于队列，还有一种“假溢出”现象，队列中尚余有足够的空间，但元素却不能入队，一般是由于队列的存储结构或操作方式的选择不当所致，可以用循环队列解决。

11.一般地，要解决队列的上溢现象可有以下几种方法：

答：可建立一个足够大的存储空间以避免溢出，但这样做往往会造成空间使用率低，浪费存储空间。

要避免出现“假溢出”现象可用以下方法解决：

第一种：采用移动元素的方法。每当有一个新元素入队，就将队列中已有的元素向队头移动一个位置，假定空余空间足够。

第二种：每当删去一个队头元素，则可依次移动队列中的元素总是使 front 指针指向队列中的第一个位置。

第三种：采用循环队列方式。将队头、队尾看作是一个首尾相接的循环队列，即用循环数组实现，此时队首仍在队尾之前，作插入和删除运算时仍遵循“先进先出”的原则。

12.如何知道循环队列是空还是满？

答：第一，采用计数器来判断，空时，计数器为0，满时，计数器为maxsize；第二，另设一个布尔变量以区别队列的空和满；第三，少用一个元素的空间，约定入队前，测试尾指针在循环意义下加1后是否等于头指针，若相等则认为队满（注意：rear 所指的单元始终为空）；

13.说明线性表,栈,队列的异同点

答：相同点:都是线性结构,都是逻辑结构的概念,都可以用顺序存储或者链表存储.栈和队列两种特殊的线性表,即受限的线性表,只是对插入,删除运算加以限制.

不同点:1 运算规则不同:线性表为随机存取.栈只允许在一端进行插入.删除运算.队列只允许在一端进行插入,另一端进行删除运算.

2 用途不同,堆栈用于子程序调用和保护现场,队列用于多道作业处理,指令存储及其他运算等等、

CSDN-南浔Pyer