1252366 蒋凌超 电子1班

# 数据结构 - 第一次讨论课

时间 2013-11-14

讨论问题

* 静态结构与动态结构的本质区别是什么？
* 对于单链表，带有头结点与不带有头结点的优缺点是什么？
* 学生成绩管理：按学号顺序输入，建立成绩表；将其按学号从大到小逆置。

可以采用哪些数据结构？如何做？算法复杂性分析。

* 医院看病排队管理

讨论人员 : 蒋凌超 林介夫 张玉龙 宋磊 杜若尘 李洪浩

讨论结果

1. 静态结构与动态结构的本质区别

这个问题在组里面有些小歧义，但是细想下来不过是组员之间没有很好的理解对方的意思罢了。小组讨论结果为静态结构和动态结构的本质区别是是否要进行内存申请。

1. 对于单链表，带有头结点与不带头结点的优缺点是什么

头结点是指放在链表最前端的用来表明链表起始位置的结点，它的数据域一般是没有意义的。

优点：总的来讲就是讲第一个结点与其他结点同化

1）由于开始结点的位置被存放在头结点的指针域中，所以对于链表的第一个位置的操作就和链表中其他位置上的操作一致，无需进行特殊处理。

2）无论链表是否为空，头指针都是指向头结点的非空指针，对于空表的处理显得更加方便

**缺点：**

1. **有很小的空间浪费**
2. **不大符合一些人的操作习惯，一般来说很多人都是习惯于第一个结点就是操作的起始点**

**3）判断链表是否为空的条件要变为head->next == NULL**

**反之可知不带头结点的优缺点**

1. 学生成绩管理 : 按学号顺序输入，建立成绩表；将其按学号从大到小逆置。

可以采用哪些数据结构?如何做?算法复杂性分析。

我们分析认为这个问题可以由我们已学过的任意一个存储结构解决，故下面一一阐

述。

1. **顺序表：**首先创建一个可以扩大容量的动态顺序表，顺序表的元素是学生信息，

故应该建立一个结构体，顺序表为结构体的一个数组。元素输入时按学号顺序

输入时，附加一个整型变量保存输入元素的个数，在逆置时，使用一个学生类

型的局部变量，则逆置可表示为 :

SeqList<student> Reverse(SeqList<student>,int n)

{

Student temp;

SeqList<student> Ltemp;

For(int i = 0;i <= n;i ++,n --)

{

SeqList<student>[i] = temp;

SeqList<student>[i] = SeqList<student>[n];

SeqList<student>[n] = temp;

}

Return Ltemp;

}

时间复杂度为 O(n)

1. **链表：**相对于顺序表，单链表的操作会稍显麻烦，因为得找到某结点得先遍历至

该结点，时间复杂度为O(n^2)；而双向链表就没有这个问题，解决起来就和顺序表差不多,时间复杂度为O(n)。

和顺序表相比，链表的优点是内存空间申请更加灵活，不会浪费。

3）**栈和队列：**这两种方法的逆置都较为容易完成，不同之处也是因为两种方法的使

用方式不同。但是事实上，个人认为用这两种方法来逆置稍显不明智，实用性不高。

1. 医院看病排队管理

我们一致认为这个问题得用队列来解决。

需要考虑的因素很多，在这里主要考虑这么几点（时间限制，仅仅讨论到的）

1. 是否有需要优先就医的人员
2. 不同科目病的医师分配

**实现过程**：我们得先建立n个队列，用来进行不同病症的出队入队，针对优先就医的病人，我们可以相应再建立n个紧急队列，当这个队列里有信息时，这个队列里的病人有更高的优先级。

对讨论课的建议

1. 鉴于我们组在课后才有机会发言的前例，希望讨论课的分组能够更少一些，或者限制每组发言时间，能给每组一个表达的机会。
2. 个人感觉讨论题必须得有讨论的价值，一些概念性的问题我觉得没有必要放在课上来讲，这个是自己在私下的时间里去弄懂的，我们应该讨论些更加贴近实际生活的问题，宁愿让题数少点，让大家能把一个综合性题目给讨论的更深些。