



# 线性表研讨课报告

着重关于医院看病排队管理

电子与信息工程学院 计算机类1班

1852409 李佳庚

## 一、对于研讨课黑板上的四个问题，现对前三个问题一一进行简要的解答。

1. 静态结构和动态结构的本质区别是什么：  
静态结构在编码时就有确定长度的连续空间。  
动态结构则不同，没有连续的空间，长度可以根据需求增加减少。
2. 对于单链表，带有头节点和不带有头节点的优缺点是什么：  
优点：带有头节点的链表可以把关于链表的操作，比如遍历、插入、删除、查找等操作化归为同一种操作方式。简化了编码。  
缺点：带有头节点的链表相对于没有头节点的链表多了一个节点，消耗了更多的空间。特别是当需要大量短链表的时候，多出来的一个头节点占用的空间很可观。
3. 学生成绩管理：按学号顺序输入、建立、插入，输出时按其学号从大到小逆置。请问可用什么数据结构，如何做，算法复杂度为多少？  
由于只需要输入、建立、插入、输出操作。并没有对查找做要求。  
再加上成绩管理需要存储的条件，更应该选择**链表**作为管理的数据结构。  
顺序输入时，利用**头插法**，这样可以做到在所有成绩录入完毕时，从头结点开始到尾节点的顺序遍历，刚好为按照学号大小逆置的顺序。  
插入建立时：  
**算法复杂度为  $O(n)$ 。**  
**输出时也为  $O(n)$ 。**
4. 医院看病排队管理：  
对这一简单的问题提出如下方面的问题：
  - a. 医院看病的真实情况如何模拟？
  - b. 医院对不同病人的分类。
  - c. 医院不同科室之间的分配。
  - d. 急诊患者与门诊患者。
  - e. 复诊患者和初诊患者。
  - f. 单个病人门诊时长。
  - g. 使用怎样的数据结构？
  - h. 各项操作的时间复杂度。
  - i. 是否符合实际？
  - j. 还能够提出怎样的问题？

## 二、医院看病排队管理的思考：

“目前在国内到医院看病，常常会有这样的感受——排队的时间常常比看病的时间要多很多！到大医院看一次病，挂号要排队，交费要排队，拿药要排队，候诊时要排队，而医生真正用于看病的时间却很短。看次病要折腾大半天时间已是常态，极端情况下甚至会出现排一天队看一分钟的情况，这不仅让普通患者感到身心疲惫，更让老人、孩子、残疾人、孕妇等患者苦不堪言。在经受了种种看病排队的烦恼之后，想必大家都已经受够了这种感觉吧。”

——《搜狐新闻：排队一天，看病一分钟，你受够了吗？》

“如果不想在凌晨五点瑟瑟发抖的冷风排队挂号，可以尝试一下晚睡一点，很多医院在夜里 12 点以后就可以挂第二天的号（各家医院不一样，可官网找到电话询问），挂好需要的号，定好第二天闹钟，安心睡醒去看病。如果不小心到达时过号了，可以找导医护士顺延几个号就可以了！如果实在挂不到号，去找您找诊区的导医护士，请主任帮您加个号，医者仁心，多半会帮您解决的！”

——《腾讯新闻：医院排队太长，想快点看病？》

综合上面两则新闻稿，我们可以看出。大医院尽管医疗资源相较于小医院而言丰富，但是面对大量患者，也是左支右绌。而根据个人经验，医院排队效率较低，除了医疗资源不充足可导致之外，本身排队当中产生的诸多问题也是原因之一。

对于患者，早上 10 点初诊的病人之前，可能会有 9 点看过病的复诊病人插队，可能有之前开药、开化验单的患者插队。甚至可能有在前几天就在网上预约过的患者排在自己前面。上述的这些插队都是生活当中合理存在的。

而对于编码而言，这些插队就好比是每个患者的入诊优先级。

复诊的病人一旦出现，就总是排在初诊的病人之前。开个化验单的患者，总是卡在复诊的病人之前。预约过的病人，一旦到了时间，并且病人在等候，就一定会无视所有条件直接开始诊断。这些都是在排队系统构建当中需要考虑到的东西。

对于优先级，我们没有学到很多。

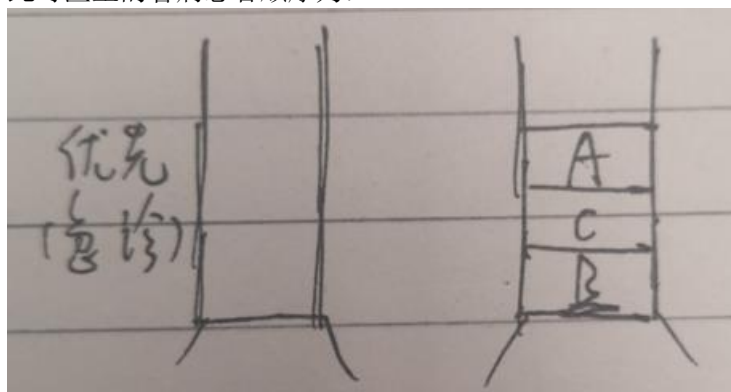
课上能够想到的，是通过建立很多个队列，来解决问题。

对每一个科室的医生，都有对应的两个队列，分别为：急诊队列和普通队列。

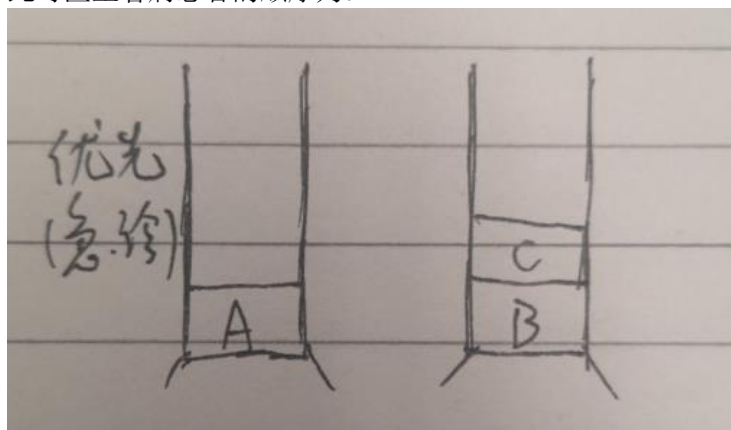
在急诊队列中，放置诸如：急诊、复诊等患者。

在普通队列中，放置初诊的患者。

当且仅当急诊队列中没有患者时，医生才能够从普通患者队列中拉取请求。（如下图）此时医生的看病患者顺序为：BCA



一旦急诊队列中有了患者，无论 B 等了多久，那么都是从 A 开始诊断。（如下图）此时医生看病患者的顺序为：ABC



当然，一个医生可能有很多优先级的患者。那么，有几个优先级，就建立几个队列，然后依次判断。这就是研讨课上同学们的想法。  
这样做，入队和出队的时间复杂度都为  $O(1)$ 。

然而我并不是这么想的。

### 三、基于最大二叉堆的优先队列

一个小医院，同时诊断患者的医生大约为 60 个左右。  
而以四川省华西医院为例，临床医生数目为 **3000 名**，门诊大夫数量十分可观。这样再给每个大夫分配 5~6 个队列，每个队列中存储的还不是简单的数据类型，而是关于患者的各项数据，所以并不实际。

很有可能造成不必要的空间浪费。

所以比如把所有的患者全部放到一个队列当中。  
如此，便可以最大化节省空间，还可以对医院等候门诊的患者数量进行快速监控。

为了完成这一目标，就不得不推出**基于最大二叉堆的优先队列**。

最大二叉堆的意思是：**任何一个父节点的值，都大于或等于它左、右孩子节点的值**。  
假设父节点的下标是 `parent`，那么它的左孩子的下标就是  $2 * parent + 1$ ，右孩子的下标就是  $2 * parent + 2$ 。（如下图）

当二叉堆插入节点时，插入位置是完全二叉树的最后一个位置，将该节点与它的父节点进行比较，如果该节点小于它的父节点，那么该与它的父节点交换位置，直到比较到堆顶位置。  
构建二叉堆就是把一个无序的完全二叉树调整为二叉堆，本质就是让所有非叶子节点依次“下沉”。时间复杂度为  $O(n)$ 。

二叉堆是实现堆排序及优先队列的基础。而我们要构造的最大优先队列，无论入队顺序如何，都是当前最大的元素优先出队。

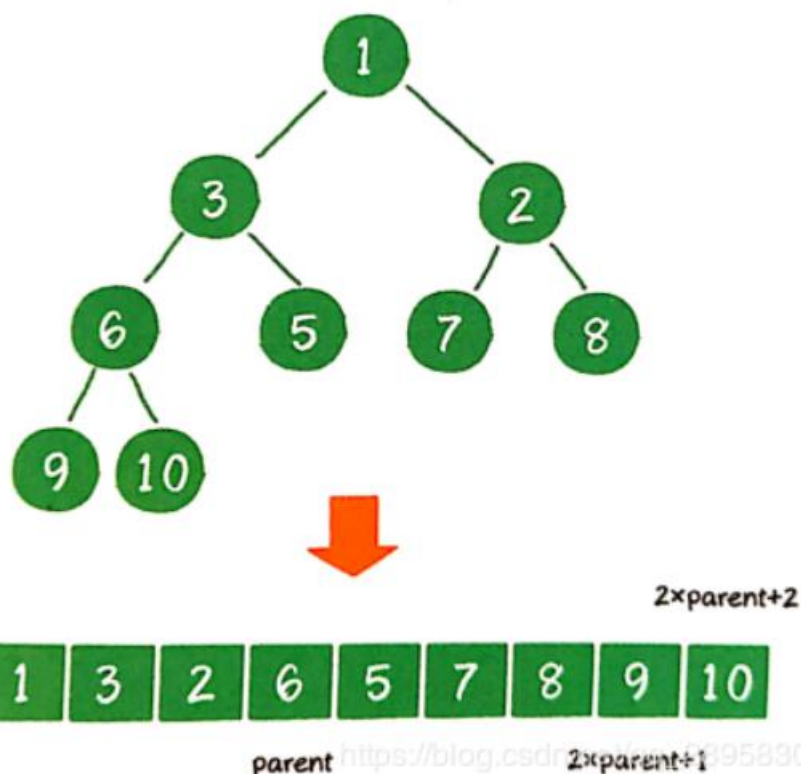
不是 First In Last Out，而是 **First In Largest Out**。

如果我们把优先级最为出队列判断的标准，那么我们就可以做到用  $O(\lg n)$  的时间找到那个元素，然后令其出队列。

装

订

线



而在医院这种实际环境中，常数的时间复杂度和对数的时间复杂度并差不了多少。不会有人因为这短暂的时间而病情加剧，所以这并不会出现什么问题。要是有个医院日接待病人为亿级，那么同时等待的病人应该也是千万级了，那个时候，才会有明显的差别。

由此。可以将所有病人放入一个等待队列当中。在查找过程中，只需要根据所挂科室筛选出所有患者，就可以通过优先级判断哪位患者该进入诊室。

比如设：

患者类型	优先级
初诊患者	0
复诊患者	1
其他患者	2
预约患者	3
急诊患者	4

挂号时间	%h:%m:%s
最短时长	2min
挂号科	t
可去科室	[x,y,z,...]

根据这些信息，就可以推断出某科的某科室的下一个患者是谁。

## 四、 其他的问题

其实还有很多功能有待实现。

比如张同学在课上讨论的那个问题：预测每名患者大概会排多少分钟的队。

再比如可否通过其他树形结构来加快查找速度。

再比如医院可能会对经常光顾医院的患者加以特殊照顾，这个时候就需要其他算法，比如 LRU 算法之类的。

## 五、对研讨课的建议

1. 我觉得研讨课这次给的问题太**过于简单**了。

前面三个问题都是课上讲过的，说老实话，第四个也没啥好说的，是个人多考虑一会儿现实情况自然就搞清如何在现学范围内找到解决办法了。

2. 研讨课并没有发挥该有的作用。

讨论问题可以加深对知识点的理解，但是由于主持人拉不下来脸终止一些完全没所谓的讨论，导致整个过程很低效，还有上台的同学，总是把注意力放在奇怪的地方，在旁人看来，根本抓不到重点。

3. 研讨课希望可以**跨班组队**。又不考勤又不积分的。这就是在暗示我来了就行。

我也更希望和另外一个班的同学交流。

不是说交流，是被指点。

另外一个班有叶神、凯哥他们，**想听听真正的大佬的发言，而不是听抓不到重点的发言。**

电子与信息工程学院 计算机类 1 班 1852409 李佳庚  
2019/10/27

装

订

线