

# 数据结构-讨论题

## 一、(1.1 什么是数据结构)

1.标题：第 1 讲讨论：对中等规模、大规模的图书摆放，你有什么更好的建议

内容：

这个问题实际上关联到如何在计算机中处理大数据的思路..... 如果你要做一个网上书城，该怎么组织图书的信息？

2.标题：第 1 讲讨论：晒一下 PrintN 在你的机器上运行的结果？

内容：

不同的机器配置会有不同的效果，这是大家发现壕的时候..... ^\_^

3.标题：第 1 讲讨论：再试一个多项式

内容：

$$f(x)=1+\sum_{i=1}^{100}x^i/i$$

给定另一个 100 阶多项式，用不同方法计算  $f(1.1)$  并且比较一下运行时间？

(这里的排版也这么丑啊..... >\_<)

4.标题：第 1 讲讨论：抽象有什么好处？

内容：

任何事物存在都要有个理由，为什么大家这么稀饭“抽象”？

## 二、(1.2 什么是算法)

标题：第 1 讲讨论：分析“二分法”

内容：

查找算法中的“二分法”是这样定义的：

给定 N 个从小到大排好序的整数序列 List[]，以及某待查找整数 X，我们的目标是找到 X 在 List 中的下标。即若有 List[i]=X，则返回 i；否则返回-1 表示没有找到。

二分法是先找到序列的中点 List[M]，与 X 进行比较，若相等则返回中点下标；否则，若 List[M]>X，则在左边的子系列中查找 X；若 List[M]<X，则在右边的子系列中查找 X。

试写出算法的伪码描述（注意要会抽象地表达哦，不要过分依赖于具体的语言和实现细节），并分析最坏、最好情况下的时间、空间复杂度。

### 三、(1.3 应用实例：最大子列和问题)

1.标题：第 1 讲讨论：算法 3 的空间复杂度是多少？

内容：

这里要考虑 2 个问题：

- 1) 由于递归而产生的空间复杂度是多少？
- 2) 算法的整体空间复杂度一共是多少？

2.标题：第 1 讲讨论：晒运行结果

内容：

在 PAT 上发布的编程题“最大子列和问题”给了非常宽松的时间上限，让大家可以至少把算法 2、3、4 分别尝试一下。另外“Maximum Subsequence Sum (25)”是 PAT(A)练习题，需要你吧算法略做修改，不仅输出最大和，而且输出相应的那个子列的首尾。

### 四、(2.1 线性表及其实现)

标题：第 2 讲讨论：链式存储中 FindKth 函数的另一种实现？

内容：

如果将链式存储中 FindKth 的函数实现（如下）做个修改：把函数最后的 if 语句判断条件改为判断 p 是否为 NULL，即：

```
if (p==NULL) return NULL;
else return p;
```

或者说直接简化为：return p;

对于这样的修改，程序还正确吗？为什么？

**List \*FindKth( int K, List \*PtrL )**

```
{ List *p = PtrL;
  int i = 1;
  while (p !=NULL && i < K){
    p = p->Next;
    i++;
  }
```

```

    if(i == K) return p;  /*找到第 K 个，返回指针*/
    else return NULL;  /* 否则返回空 */
}

```

## 五、(2.2 堆栈)

标题：第 2 讲讨论：堆栈顺序存储的另一种实现？

内容：

有人给出了堆栈用数组实现的另一种方式，即直接在函数参数中传递数组和 `top` 变量（而不是两者组成的结构指针），其中 `Push` 操作函数设计如下。这个 `Push` 函数正确吗？为什么？

```

#define MaxSize 100

ElementType S[MaxSize];

int top;

void Push(ElementType *S, int top, ElementType item)
{
    if (top==MaxSize-1) {
        printf("堆栈满"); return;
    }else {
        S[++top] = item;
        return;
    }
}

```

## 六、(2.3 队列)

标题：第 2 讲讨论：如何用两个堆栈模拟实现一个队列？

内容：

如何用两个堆栈模拟实现一个队列？如果这两个堆栈的容量分别是  $m$  和  $n$  ( $m > n$ )，你的方法能保证的队列容量是多少？

## 七、(2.4 应用实例：多项式加法运算)

标题：第 2 讲讨论：日常生活中的堆栈和队列

内容：

日常生活中队列很常见。还有什么堆栈的典型例子？

夏天逛超市时，你可能会忍不住想喝一瓶冰饮料降降温。可是，困扰你的是：放冰箱外头的饮料往往并不冰，而冰箱深处你够不着的地方，才是你想要的。

想想看：冰箱中的饮料应该组织成“队列”还是“堆栈”？你对超市中的冰箱设计有什么改进性的建议？

## 八、(3.1 树与树的表示)

1.标题：第 3 讲讨论：黄金分割查找？

内容：

在二分查找中，我们是取  $mid$  等于  $left$  和  $right$  的中间值，即用等分的方法进行查找。

那为什么一定要等分呐？能不能进行“黄金分割”？也就是  $mid=left+0.618(right-left)$ ，当然  $mid$  要取整数。如果这样查找，时间复杂性是多少？也许你还可以编程做个试验，比较一下二分法和“黄金分割”法的执行效率。

2.标题：第 3 讲讨论：森林及表示

内容：

树的集合称为森林。是否也可以使用“儿子-兄弟”表示法存储森林？如何实现？

## 九、(3.2 二叉树及储存结构)

标题：第 3 讲讨论： $m$  叉树中各类结点数之间的关系

内容：

在二叉树中，我们知道叶结点总数  $n_0$  与有两个儿子的结点总数  $n_2$  之间的关系是： $n_0=n_2+1$ 。

那么类似关系是否可以推广到  $m$  叉树中？也就是，如果在  $m$  叉树中，叶结点总数是  $n_0$ ，有一个儿子的结点总数是  $n_1$ ，有 2 个儿子的结点总数是  $n_2$ ，有 3 个儿子的结点总数是  $n_3$ ，...

那么， $n_i$  之间存在什么关系？

## 十、(3.3 二叉树的)

1.标题：第 3 讲讨论：如何用堆栈实现后序遍历的非递归程序

内容：

我们前面看到，借助堆栈可以实现前序遍历、中序遍历的非递归程序，而且两者的程序结构几乎一样。

那么，是否也可以借助堆栈实现后序遍历的非递归程序？是不是挪动一下 `printf` 语句就可以了？

2.标题：第 3 讲讨论：将层序遍历中的队列改为堆栈

内容：

如果将层序遍历中的队列改为堆栈，是否也是一种树的遍历？可以应用这种方法改造出一种前序、中序、后序的非递归遍历吗？

### 十一、(4.1 二叉搜索树)

标题：第 4 讲讨论：搜索树比较序列是否合理的判别

内容：

如果有人跟你说，在某棵搜索树上找 63 的过程是：先比较根 39，然后比较根的右儿子 101，接着比较 101 的左儿子 25，再比较 25 的右儿子 63，这样就找到了。

也就是说，他在搜索树查找时的比较序列是：39，101，25，63。问：这样的查找序列可能吗？你能否设计一个算法或者程序，判别给定的查找序列是否合理？

### 十二、(4.2 平衡二叉树)

标题：第 4 讲讨论：是否可以用左右子树结点数差来衡量二叉树是否平衡？

内容：

1962 年，G.M. Adelson-Velsky 和 E.M. Landis 发表了一篇论文 "An algorithm for the organization of information"，提出了一种自平衡的二叉树，后人以他们的名字命名为 AVL 树。在 AVL 树中，评价二叉树是否平衡，衡量的标准是左右子树之间的高度差，即平衡因子。

为什么一定要用高度差来衡量呢？用左右两边结点数差来衡量是否也可行？比如，是否可以将“平衡因子”考虑为：左子树（右子树）结点数不能超过右子树（左子树）结点数一倍以上？

记  $N_l$  为左子树结点数， $N_r$  为右子树结点数。如果对所有结点都有： $1/3 \leq N_l/(N_l+N_r) \leq 2/3$ ， $1/3 \leq N_r/(N_l+N_r) \leq 2/3$ ，则认为该二叉树是“平衡”的。

这样的“平衡”树在高度方面是否也是  $O(\log n)$ ？如果发现不平衡，调整方便不方便？

大家可以思考一下。如果有好的方案，说不定在计算机历史上又多了一种以你名字命名的 xxx 平衡树。

### 十三、(5.1 堆)

标题：第 5 讲讨论：减堆中某个元素值的操作如何实现？

内容：

如果想在堆 H 上实现一种新的操作：DecreaseKey(H,P,X)，将堆 H 中，位于 P 的元素的值减去 X。这个操作如何实现？当然，要求操作执行后，H 仍然是个堆，堆中的元素允许移动。

#### 十四、(5.2 哈夫曼树与哈夫曼)

标题：第 5 讲讨论：判别是否是前缀码的算法

内容：

是否可以设计一个算法，判别某种编码方案是否是前缀码？

比如，这样的编码方案是前缀码吗：00，01，10，11，101。

#### 十五、(5.3 集合与运算)

标题：第 5 讲讨论：如何让并查树更矮？

内容：

在用于实现并查集的树（并查树）中，每个结点关心的是父结点在哪里，因此树中每个结点只要有个 `parent` 值指向父结点就可以方便找到根结点了，而且也不要求是二叉树。当然，树的高度仍然是影响 `find` 操作效率的关键因素。

如果我们经常要做 `find` 操作，你有什么办法可以降低树的高度使 `find` 的效率更高？

#### 十六、(6.1 什么是图)

标题：第 6 讲讨论：关于用链表得到有向图的出、入度--其他办法

内容：

关于用链表表示图并且能方便地得到有向图的出、入度，你有什么其他的办法吗？

#### 十七、(6.2 图的遍历)

1.标题：第 6 讲讨论：把迷宫出口换到哪里就该 `BFS` 不爽了？

内容：如题

2.标题：第 6 讲讨论：试比较 `DFS` 和 `BFS` 的优点和缺点

内容：

它们分别适合解决什么样的问题？包括空间效率和时间效率。在什么情况下用哪种比较好？

#### 十八、(6.3 应用实例：拯救 007)

标题：第 6 讲讨论：有必要存储图吗？

内容：

在 007 的问题中，我们是否有必要把问题中的图按照邻接矩阵或者邻接表的形式存储？

为什么？

## 十九、(6.4 应用实例：六度空间)

标题：第 6 讲讨论：用什么方法存储图比较好？为什么？

内容：

在六度空间问题中，题目假设边的条数不超过节点数的 33 倍。用什么方法存储图比较好？为什么？

## 二十、(7 树之习题选讲)

标题：树习题讨论：最小  $X$  为什么不是 1？

内容：

视频中说：要使  $L = 2^{H-1} - 1 + X$  得到正确结果， $X$  能取的最小值是 0。为什么不是 1 呢？

## 二十一、(7.1 最短路径问题)

1.标题：第 7 讲讨论：无权图的单源最短路初始化

内容：如果 `dist[]` 全部初始化为 0，会有什么问题？

2.标题：第 7 讲讨论：无权图的单源最短路输出

内容：

如何顺序输出  $S$  到  $W$  的路径上的所有顶点？

3.标题：第 7 讲讨论：Floyd 算法与负值圈

内容：1) 如果图中有负值圈，Floyd 算法还能用吗？

2) 如何知道图中是否存在负值圈？

## 二十二、(8.1 最小生成树问题)

标题：第 8 讲讨论：试举一个最小生成树不唯一的例子

内容：最好画个图 —— 无图无真相呀~

## 二十三、(8.2 拓扑排序)

标题：第 8 讲讨论：容器的影响

内容：

分别用队列和堆栈作为容器，对计算机专业课程进行拓扑排序，得到的序列有什么区别？用哪种容器排课更合理？

## 二十四、(9.3 堆排序)

标题：第 9 讲讨论题：堆排序最适合解决什么样的问题？

内容：最好举个例子说明你的观点。

## 二十五、(9.4 归并排序)

标题：第 9 讲讨论：晒运行结果

内容：

PAT 上给出了一道实验题，并且给出了每组数据的特点描述。建议大家把今天讲到的所有算法实现一遍，比较它们对于不同特点的数据进行排序的效率，从而对不同的排序算法加深感性认识。

## 二十六、(10.1 快速排序)

标题：第 10 讲讨论题：晒快排实现结果

内容：1) 对 pivot 的 3 种不同选择办法：直接选  $A[0]$ 、用随机函数、三元选中值，分别实现快速排序算法，在此晒一下运行的效率。

2) 对 Cutoff 的 5 种不同取值：50、100、200、300、400，分别实现快速排序算法，在此晒一下运行的效率。

## 二十七、(10.3 基数排序)

标题：第 10 讲讨论题：LSD 任何时候都比 MSD 快吗？

内容：详细比较一下各个参数对它们的时间复杂度的影响。

## 二十八、(10.4 排序算法的比较)

标题：第 10 讲讨论题：继续晒结果

内容：请继续利用 09-排序 1 将新学的各种排序算法的运行效率进行比较。

## 二十九、(11.2 散列函数的构造方法)

标题：第 11 讲讨论：在除留余数法中,为什么  $p$  要取素数？

内容：

在除留余数法中， $H(key)=key \bmod p$ ，为什么  $p$  要取素数？ 你可以举些例子分析一下。

## 三十、(11.3 冲突处理方法)

标题：第 11 讲讨论：已知散列表状态，推测可能的元素输入顺序

内容：



下面图示的内容是课程中提到的例子，即给出一个序列，按顺序逐个插入散列表中.图中显示了散列表最后的状态。

我们可以思考这样一个逆问题：如果已知散列表的状态，我们能不能推测出可能的输入序列顺序？（可能会有多个这样的序列顺序）

这是一个很好玩的问题。

想明白了，就可以到 pat 系统中去练练。

**【例】** 将acos、define、float、exp、char、atan、ceil、floor，  
顺次存入一张大小为26的散列表中。

$H(key)=key[0]-'a'$ ，采用线性探测 $d_i=i$ 。

acos	atan	char	define	exp	float	ceil	floor		.....	
0	1	2	3	4	5	6	7	8		25

### 三十一、(11.4 散列表的性能)

标题：第 11 讲讨论：以 QQ 账号的申请和登录为例，晒晒运行结果

内容：

散列查找的效率到底好不好，还是需要动手试试。本周的第二个程序作业题是“QQ 帐户的申请与登录”问题，

你可以去试试，并且可以将散列方法与排序方法，甚至 AVL 树查找方法做个比较。