

数据结构应用题参考答案

咸鱼注：关注一下我们的微博，今后会有更多的内容分享，@王道咸鱼老师 @王道楼楼老师 @王道计算机考研备考

栈、队列、数组的应用

41. (10 分) 请回答以下问题：

(1) 队列在顺序存储时的“假溢出”现象指什么？

(2) 简述一种可行的假溢出的解决方法。

(3) 若用数组 $q[1..m]$ 表示队列，队列头指针 $front$ 、尾指针 $rear$ 的初值均为 1，基于 (2) 中的方法，如何求队列的当前长度？如何判定队空？如何判定队满？

考点：队列的应用

来源：北京邮电大学803 2018年

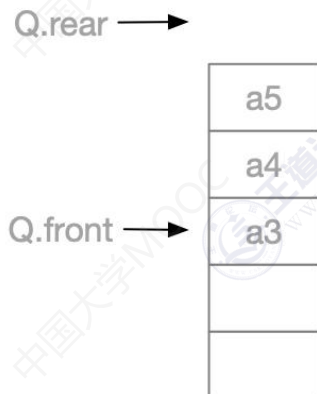
【参考答案】

(1)

对于顺序存储的队列，执行入队和出队操作时，头、尾指针只增大不减小，致使出队元素的空间无法被重新利用。因此，尽管队列中实际元素个数可能远小于数组大小，但可能由于尾指针已超出数组下标的界限而不能执行入队操作。该现象称为“假溢出”。

(咸鱼注：有的简答题，如果你觉得文字表述不够清晰，可能会导致老师扣你分，那么可以主动配上示例图，比如加上后边这段...当然，如果你觉得文字表达已经OK了，完全可以不画图)

如下图所示，是数组大小为5的顺序队列，队尾指针超出了数组范围，尽管此时仍有未使用的空间，但无法入队，此时发生“假溢出”。



(2)

解决假溢出的方法，可以采用循环队列。入队操作时，若队列不满，则将新元素插入队尾指针 rear 所指位置，并令 $rear = (rear+1)\%m$ ；出队操作时，若队列不空，则令队头指针 front 所指元素出队，并令 $front = (front+1)\%m$ 。其中，m 为队列总长度。

(3)

队空条件：front==rear

队满条件：front==(rear+1)%m

队列长度：length = (rear-front+m)%m

树和二叉树的应用

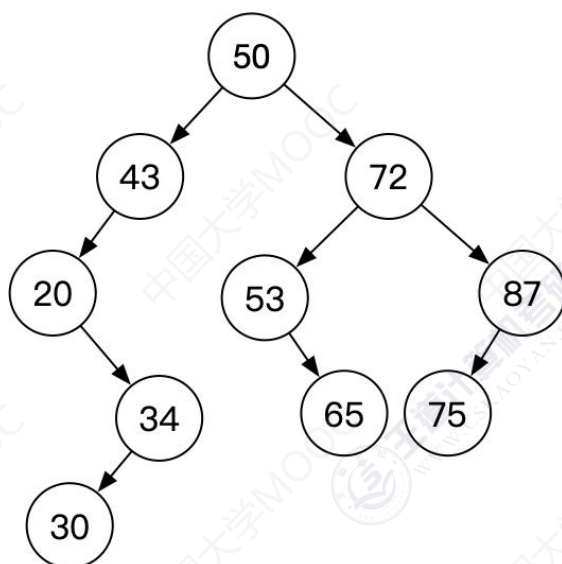
二、利用逐点插入法建立序列 (50, 72, 43, 87, 75, 20, 34, 53, 65, 30) 对应的二叉排序树。并求出这 10 个元素基于该二叉排序树的组织，在等概率情况下查找成功时的平均查找长度。(15 分)

考点：二叉排序树

来源：华中科技大学834 2018年

【参考答案】

二叉排序树中，各结点关键字的大小关系为：左子树≤根节点≤右子树。依次插入10个元素后，二叉排序树的结构如下图所示：



若查找的目标关键字位于树的第 i 层，则查找长度为 i。因此在等概率情况下，查找成功的平均查找长度：

$$ASL_{\text{成功}} = (1 \times 1 + 2 \times 2 + 3 \times 3 + 4 \times 3 + 5 \times 1) / 10 = 3.1$$

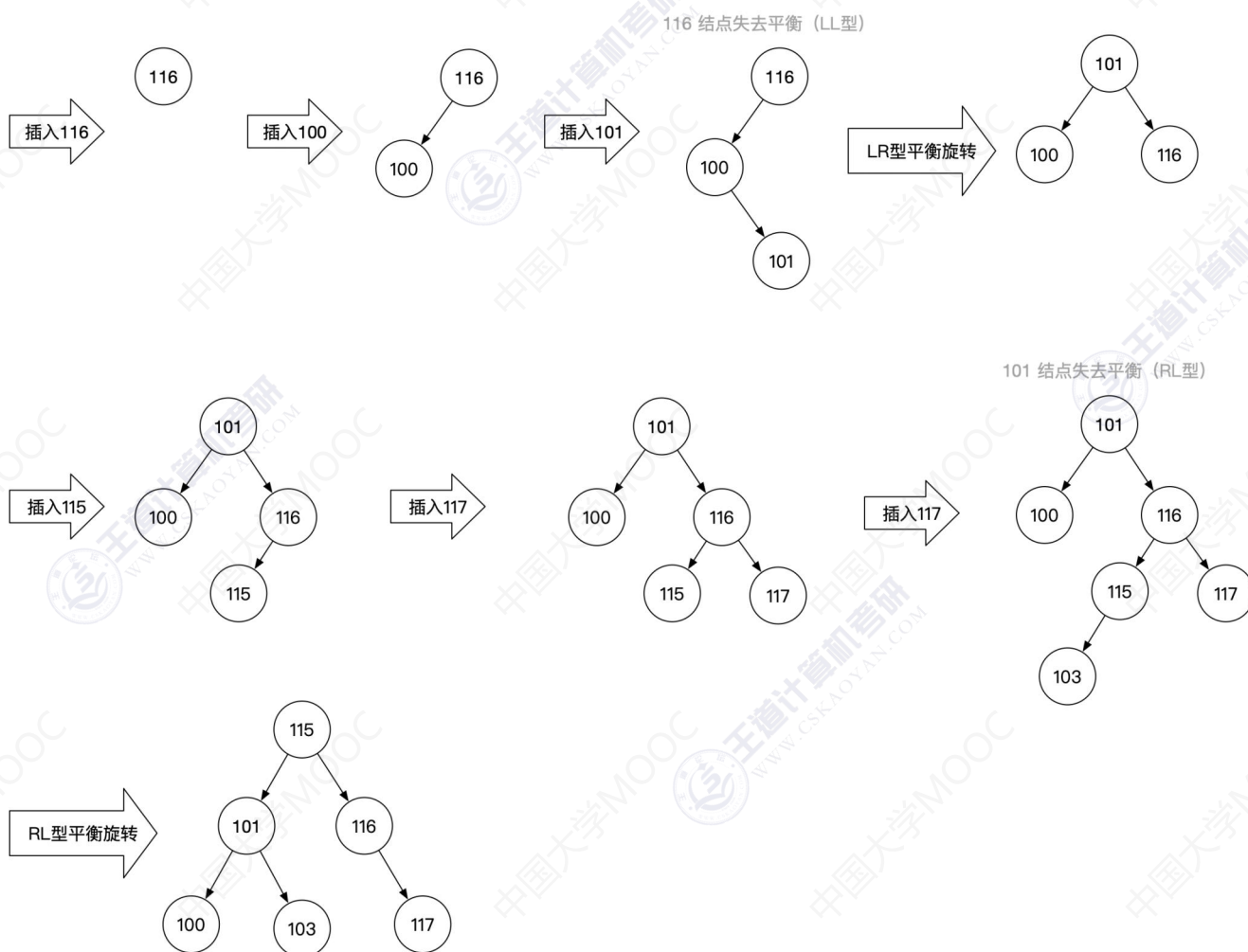
47. 将关键字序列{116, 100, 101, 115, 117, 103}依次插入到初始为空的平衡二叉树 (AVL树), 给出每插入一个关键字后的平衡树, 并说明其中可能包含的平衡调整步骤 (即, 先说明是哪个结点失去平衡, 然后说明做了什么平衡处理); 然后分别给出前序、中序和后序遍历该二叉树的输出结果。
(10分)

考点: 平衡二叉树

来源: 中国科学院大学863 2019年

【参考答案】

各关键字的插入过程如下所示:



对该二叉树的遍历结果如下

前序遍历: 115, 101, 100, 103, 116, 117

中序遍历: 100, 101, 103, 115, 116, 117

后序遍历: 100, 103, 101, 117, 116, 115

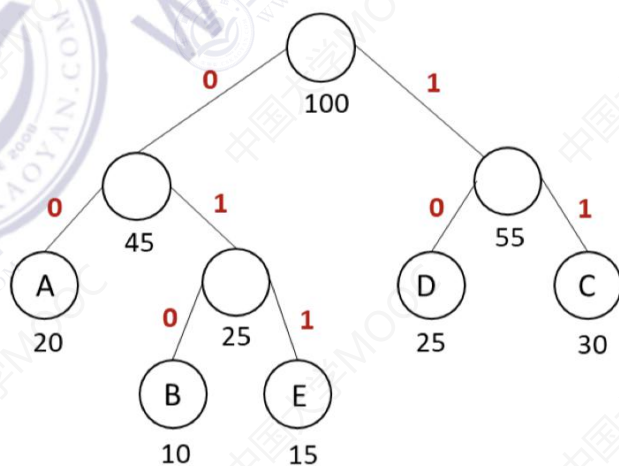
3. (10 分) 假设传输的数据包含 5 个字母 A,B,C,D,E, 字母出现的频率: A-20%, B-10%, C-30%, D-25%, E-15%, 请构造哈夫曼编码以及哈夫曼树。

考点: 哈夫曼树和哈夫曼编码

来源: 北京大学869 2019年

【参考答案】

构造的哈夫曼树为:



哈夫曼编码为:

A: 00 B: 010 C: 011 D: 10 E: 11

(咸鱼注: 本题哈夫曼树、哈夫曼编码不唯一, 只要合理都可得分)

(2) 具有哪些特征的编码是一定会有的

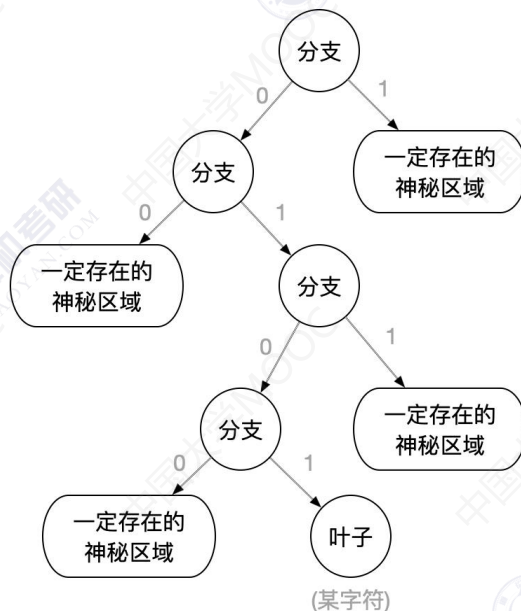
来源：四川大学874 2018年

(2) 以 1, 00, 011, 0100 为开头的编码一定会有。

```

graph TD
    A((分支)) -- 0 --> B((分支))
    A -- 1 --> C((分支))
    B -- 0 --> D((分支))
    B -- 1 --> E((分支))
    C -- 0 --> F((分支))
    C -- 1 --> G((叶子))
  
```

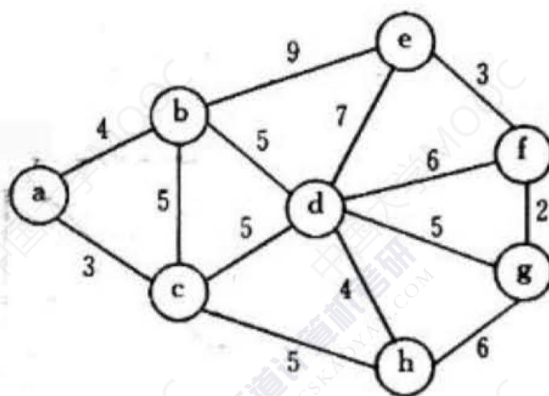
咸鱼注2：要注意哈夫曼树的特性——所有分支节点的度必然是2，因此上图中，各个分支节点的另一侧肯定会挂有其他节点。像这个亚子：



因此，以 1, 00, 011, 0100 为开头的编码一定会存在。

图的基本应用

2. (15 分) 针对下图的无向带权图，



- (1) 写出它的邻接矩阵，并按 Prim 算法求其最小生成树。
- (2) 写出它的邻接表，并按 kruskal 算法求其最小生成树。

考点：图的存储、最小生成树

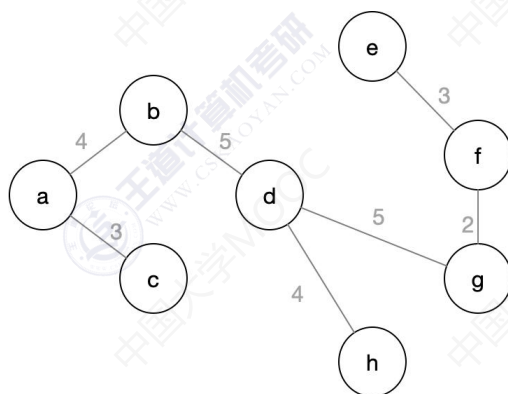
来源：北京大学869 2018年

【参考答案】

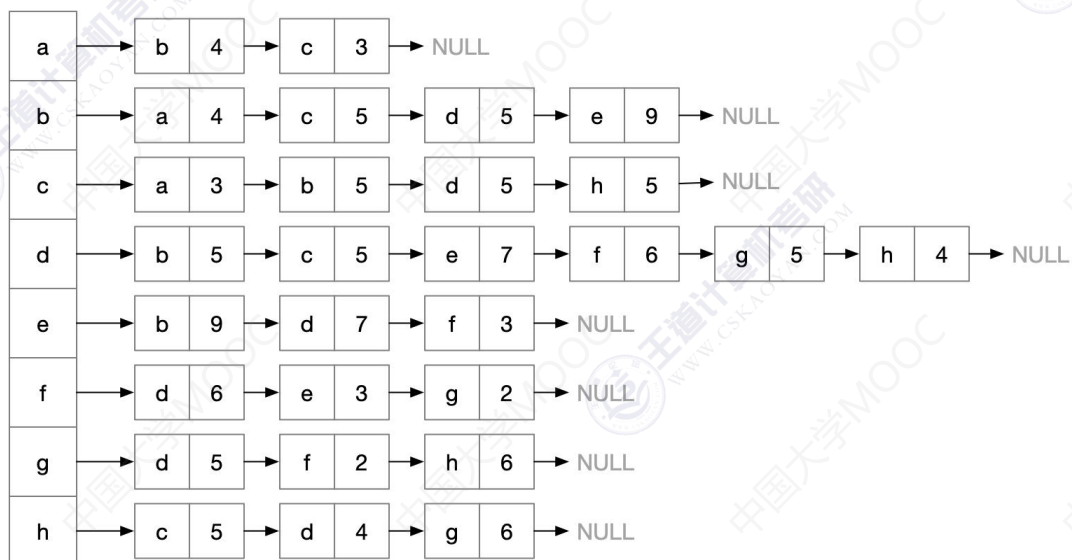
(1) 邻接矩阵如下:

	a	b	c	d	e	f	g	h
a	0	4	3	∞	∞	∞	∞	∞
b	4	0	5	5	9	∞	∞	∞
c	3	5	0	5	∞	∞	∞	5
d	∞	5	5	0	7	6	5	4
e	∞	9	∞	7	0	3	∞	∞
f	∞	∞	∞	6	3	0	2	∞
g	∞	∞	∞	5	∞	2	0	6
h	∞	∞	5	4	∞	∞	6	0

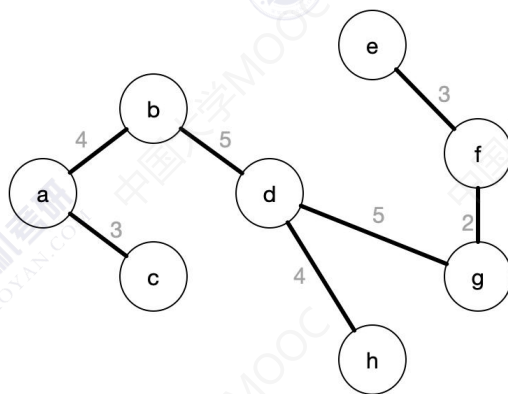
按 Prim 算法得到的最小生成树如下:



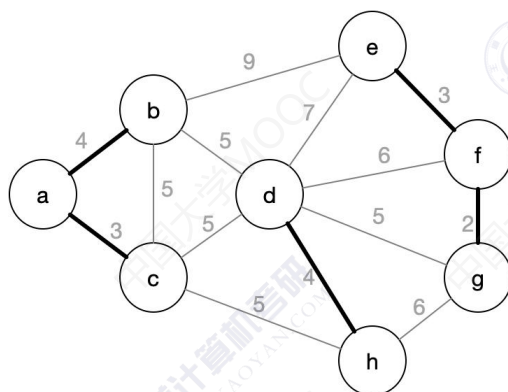
(2) 邻接表如下:



按 Kruskal 算法得到的最小生成树如下:



咸鱼注：大家在手动模拟 Kruskal 算法执行时，估计会在下面这一步遇到这个问题。克鲁斯卡尔算法每次要选择一条权值最小的边，你肯定可以选出权值为2、3、4的这几条边，但是，接下来呢？？？一脸懵逼对吧，接下来好几条边的权值都是5， $\langle c,d \rangle < \langle b,d \rangle < \langle d,g \rangle < \langle c,h \rangle$ 那应该优先选哪条呢？



先说结论：这几条边的优先级依次是 $\langle b,d \rangle < \langle c,d \rangle < \langle c,h \rangle < \langle d,g \rangle$ 。看出规律了吧？优先看第一个字母（起点）哪个更小，第一个字母相同时再看第二个字母（终点）哪个更小。

想知道具体原因，可以回去看看基础课跟大家说过的克鲁斯卡尔算法的实现原理。首先，预处理阶段要将所有的边按权值递减排序。各条边被选择的优先级，和预处理阶段的排序结果是一致的。那大家思考一下，采用邻接表保存的图，要对各条边进行排序，如果是你，你会怎么做呢？

可以使用插入排序的策略。依次摘下顶点a的各个边结点，并插入到有序序列中；再依次摘下顶点b的各个边结点，并插入到有序序列中...以此类推。因此起点越小的边，排序之后越靠前，优先级也越高。如果想不明白具体原理，那记住结论也行。relax~

二、问答题：（题号41~47，共70分）

46. 已知图G的邻接矩阵为
请画出该图，并求该图G的关键
路径及路径长度。（10分）

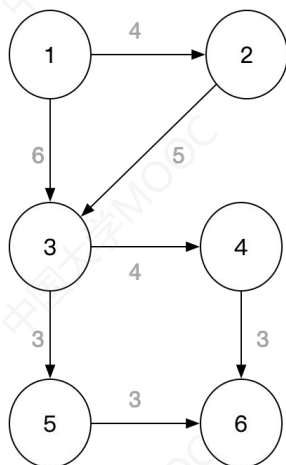
0	4	6	∞	∞	∞
∞	0	5	∞	∞	∞
∞	∞	0	4	3	∞
∞	∞	∞	0	∞	3
∞	∞	∞	∞	0	3
∞	∞	∞	∞	∞	0

考点：关键路径

来源：中国科学院大学863 2017年

【参考答案】

根据邻接矩阵画出该有向图如下：



则有如下表格：

求所有事件的最早发生时间 $ve(i)$ 和最迟发生时间 $vl(i)$

	1	2	3	4	5	6
$ve(i)$	0	4	9	13	12	16
$vl(i)$	0	4	9	13	13	16

求所有活动的最早开始时间和最迟开始时间以及这两个时间的差值

	a12	a13	a23	a34	a35	a46	a56
$e(i)$	0	0	4	9	9	13	12
$l(i)$	0	3	4	9	10	13	13
$d = l - e$	0	3	0	0	1	0	1

所有 $d=0$ 的活动构成关键路径，即 a12, a23, a34, a46 是关键路径的边，对应路径为 {1,2,3,4,6}，关键路径长度为 $4+5+4+3 = 16$ 。

咸鱼注：对于408，如果题目没有明确要求你写出求解过程，则可以直接回答答案即可。对于本题，可以用更快的方法得到关键路径——关键路径就是从源点到汇点的所有可能路径中，最长的一条。知道这个技巧，就能快速看出最长的一条路径，1，2，3，4，6

2. (15 分) 一个有六个顶点的有向带权图，其邻接矩阵 A 为上三角矩阵，存储方式为行优先的数组存储，数组元素为 4,6, ∞ , ∞ , ∞ ,5, ∞ , ∞ , ∞ ,4,3, ∞ , ∞ ,3,3

(1) 请画出该图的邻接矩阵

(2) 根据邻接矩阵画出有向图

(3) 计算关键路径

考点：数组的应用、矩阵压缩存储、图的存储、关键路径

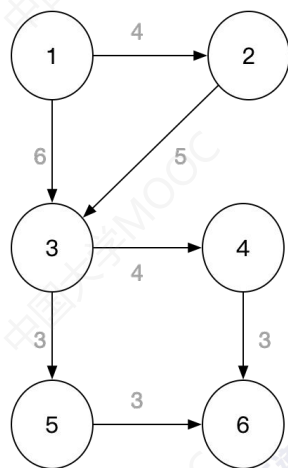
来源：北京大学869 2016年

【参考答案】

(1) 邻接矩阵如下：

$$A = \begin{bmatrix} 0 & 4 & 6 & \infty & \infty & \infty \\ \infty & 0 & 5 & \infty & \infty & \infty \\ \infty & \infty & 0 & 4 & 3 & \infty \\ \infty & \infty & \infty & 0 & \infty & 3 \\ \infty & \infty & \infty & \infty & 0 & 3 \\ \infty & \infty & \infty & \infty & \infty & 0 \end{bmatrix}$$

(2) 假设6个顶点的编号分别为1~6，则有向图如下：



(3) 关键路径为：1, 2, 3, 4, 6。关键路径长度为4+5+4+3=16。

咸鱼注：北大16年的这个真题，对比上面那个国科大17年真题，你发现了什么惊天大秘密？所以说，数据结构应用题的题型还挺难创新的，很多命题老师都是把某些例子基本条件修修改改，就变成真题了。手动摊手，ㄟ(ˉ▽ˉ)ㄟ。。。所以，根据强化课给大家指出的应用题命题重点，多做王道书相应部分的课后习题，是一种高效的备考策略。找准方向，冲就对了。

41. (10 分) 已知有向图描述为 $\{ \langle A, B, 1 \rangle, \langle A, E, 6 \rangle, \langle A, D, 7 \rangle, \langle B, C, 2 \rangle, \langle B, E, 4 \rangle, \langle C, D, 3 \rangle, \langle C, E, 1 \rangle, \langle E, D, 1 \rangle \}$ ，各项中的数字表示两顶点间的权值。

(1) 画出该有向图；

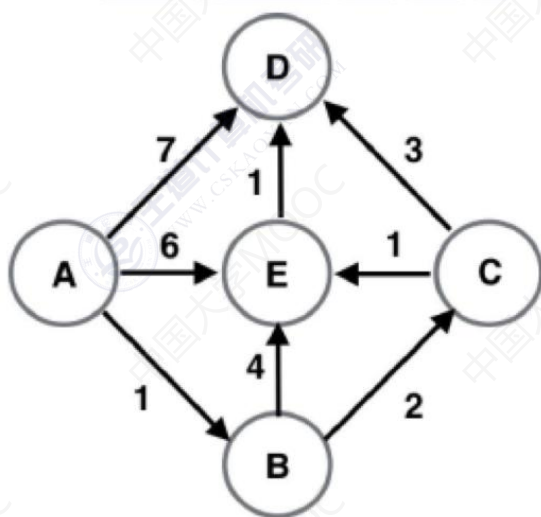
(2) 利用迪杰斯特拉 (Dijkstra) 算法求顶点 A 到其它各顶点间的最短距离，写出求解过程。

考点：最短路径

来源：北京邮电大学803 2017年

【参考答案】

(1) 您请看：



(2) Dijkstra 算法求A到其他节点的最短路径过程如下：

顶点	第 1 趟	第 2 趟	第 3 趟	第 4 趟
B	1 A→B			
C	无穷	3 A→B→C		
D	7 A→D	7 A→D	6 A→B→C→D	5 A→B→C→E→D
E	6 A→E	5 A→B→E	4 A→B→C→E	
集合 S	{A,B}	{A,B,C}	{A,B,C,E}	{A,B,C,E,D}

因此

A到B的最短路径为A→B，路径长度为1；

A到C的最短路径为A→B→C，路径长度为3；

A到D的最短路径为A→B→C→E→D，路径长度为5；

A到E的最短路径为A→B→C→E，路径长度为4；

咸鱼注：如果408中考察DJ算法，不太可能直接考代码。但是题目有可能会要求你写出求解过程，因此建议大家要对DJ算法这个表的画法要熟悉，因为你用其他方式可能很难描述清楚该算法的执行过程。除了平时多注意训练，上考场前也可以拿出来迅速看一眼，防止遗忘。对于这种很容易忘记的内容，建议大家为自己写一个“Check-List”，简要标记你经常忘记的内容，在考试前几天会很有用，让你的复习更有方向

查找算法的分析和应用

2. 采用哈希函数 $H(k) = 3 * k \bmod 13$ 并用线性探测开放地址法处理冲突，在散列地址空间 $[0..12]$ 中对关键字序列 22, 41, 53, 46, 30, 13, 1, 67, 51。

(1) 构造哈希表（画示意图）；

(2) 装填因子；

(3) 成功的平均查找长度。

(4) 不成功的平均查找长度。

考点：查找算法的分析与应用（散列查找）

来源：华中科技大学834 2017年

【参考答案】

(1) 哈希表示意图如下：

散列地址	0	1	2	3	4	5	6	7	8	9	10	11	12
关键字	13	22		53	1		41	67	46		51		30

(2) 散列表的地址空间大小为13，有9个关键字已经插入，因此装填因子 $= 9/13$

(3) 查找成功时的平均查找长度 $ASL_{成功} = (1 \times 7 + 2 \times 2) / 9 = 11/9$

咸鱼注：

对于关键字13， $3 * 13 \% 13 = 0$ ，查找长度为1

对于关键字22， $3 * 22 \% 13 = 1$ ，查找长度为1

对于关键字53， $3 * 53 \% 13 = 3$ ，查找长度为1

对于关键字1， $3 * 1 \% 13 = 3$ ，查找长度为2

对于关键字41, $3 \times 41 \% 13 = 6$, 查找长度为1

对于关键字67, $3 \times 67 \% 13 = 6$, 查找长度为2

对于关键字46, $3 \times 46 \% 13 = 8$, 查找长度为1

对于关键字51, $3 \times 51 \% 13 = 10$, 查找长度为1

对于关键字30, $3 \times 30 \% 13 = 12$, 查找长度为1

因此查找成功时的平均查找长度 $ASL_{成功} = (1 \times 7 + 2 \times 2) / 9 = 11 / 9$

(4) 查找失败时的平均查找长度 $ASL_{失败} = (3 + 2 + 1 + 3 + 2 + 1 + 4 + 3 + 2 + 1 + 2 + 1 + 4) / 13 = 29 / 13$

咸鱼注:

对于散列地址为0的关键字, 查找失败时的查找长度为3。(发生冲突时, 通过线性探测, 找到第一个空, 也就是2的位置, 才能确定查找失败)

对于散列地址为1的关键字, 查找失败时的查找长度为2。(发生冲突时, 通过线性探测, 找到第一个空, 也就是2的位置, 才能确定查找失败)

对于散列地址为2的关键字, 查找失败时的查找长度为1。(刚开始检查的Hash地址就是空的, 可以直接确定查找失败)

其他分析方法类似。需要注意, 若散列表采用这种顺序存储, 则对空结点的检查也算一次关键字的对比。而对于采用拉链法实现的散列表, 对空指针的比较不能算作一次关键字对比。(忘记的同学回去看视频or课件)

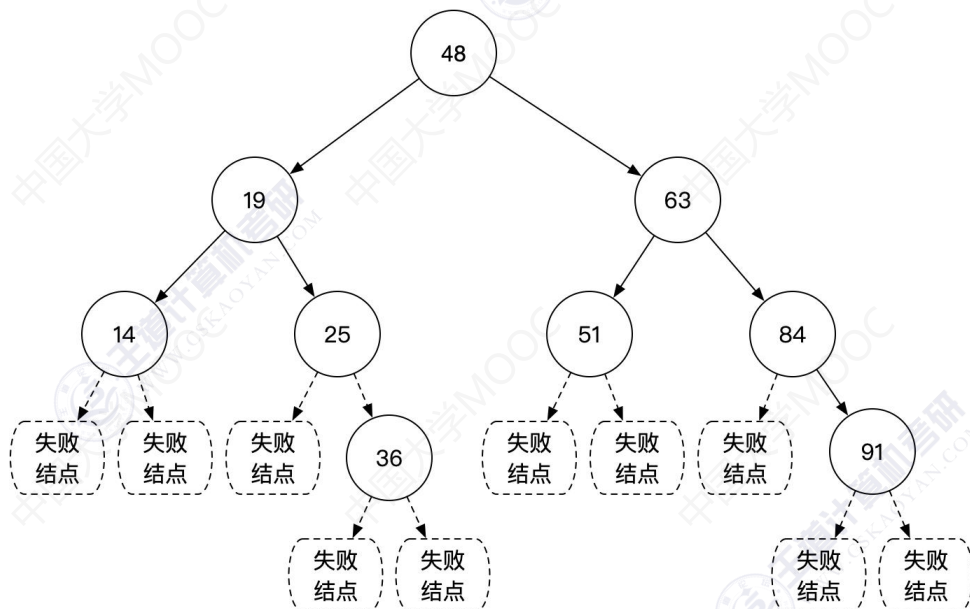
3. 设一组有序的记录关键字序列为 (14, 19, 25, 36, 48, 51, 63, 84, 91), 运用二分法进行查找, 请给出二分查找的判定树, 以及查找关键字 84 时的比较次数, 并计算出查找成功时的平均查找长度。

考点: 查找算法的分析与应用 (二分查找)

来源: 华中科技大学834 2019年

【参考答案】

二分查找的判断树如下:



查找成功时的平均查找长度 $ASL_{成功} = (1 + 2 \times 2 + 3 \times 4 + 4 \times 2) / 9 = 25 / 9$

排序算法的应用

31. (10 分) 序列{24, 4, 32, 55, 62, 18, 32*, 39, 13, 35}, 写出应用下列排序算法进行第一趟排序之后的序列状态。

- (1) 直接插入排序
- (2) 冒泡排序
- (3) 快速排序
- (4) 简单选择排序
- (5) 二路归并排序

考点：排序算法的应用（各种排序）

来源：北京大学869 2020年

【参考答案】

- (1) 4 24 32 55 62 18 32* 39 13 35
- (2) 4 24 32 55 18 32* 39 13 35 62
- (3) 13 4 18 24 62 55 32* 39 32 35
- (4) 4 24 32 55 62 18 32* 39 13 35
- (5) 4 24 32 55 18 62 32* 39 13 35

咸鱼注：此题题干漏了个条件：按元素递增排序

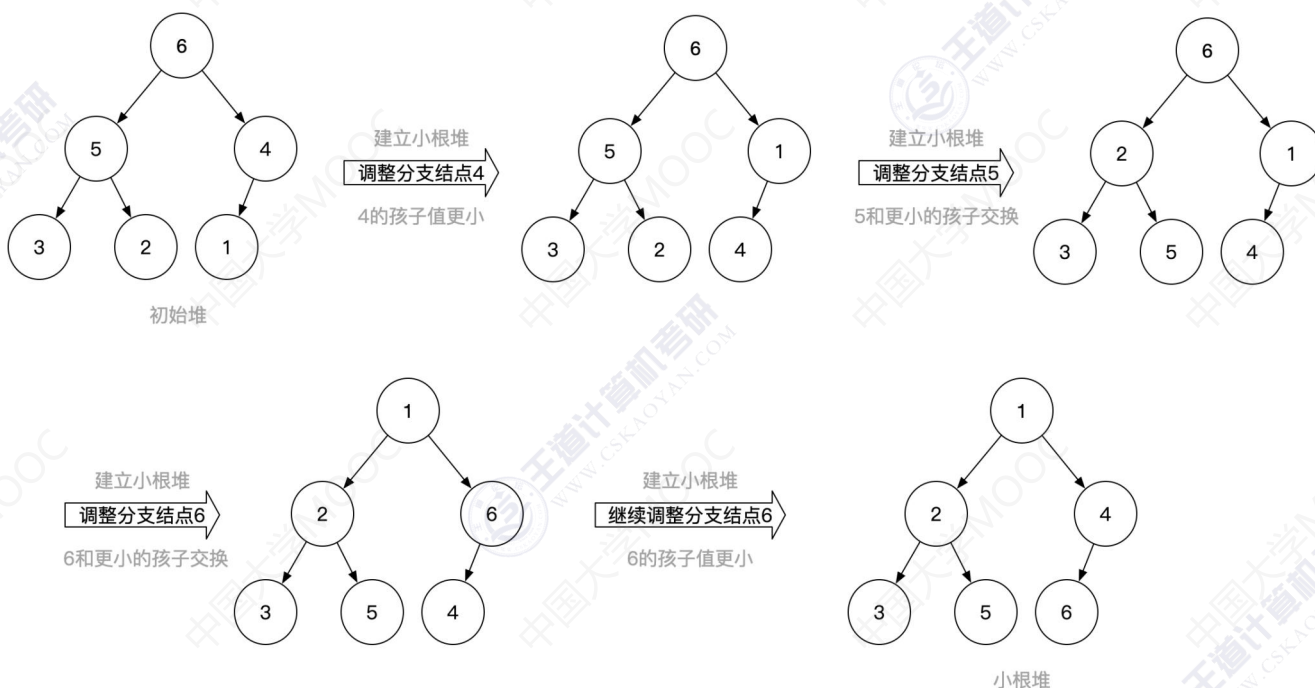
3. {6, 5, 4, 3, 2, 1} 利用数组建成一个最小堆并使用堆排序将其排序成唯一的降序数组。要求画出所有中间过程。

考点：排序算法的应用（堆排序）

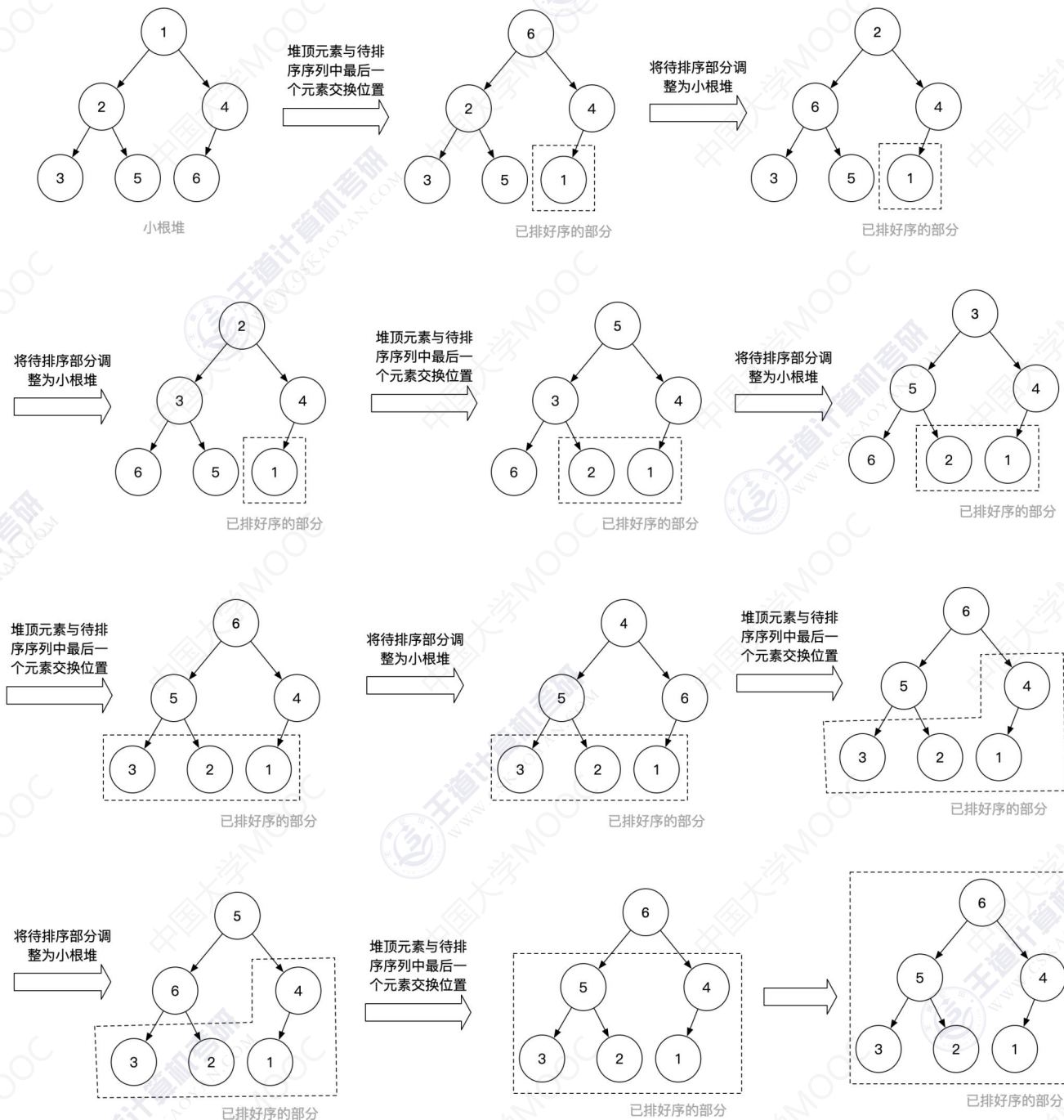
来源：华中科技大学834 2017年

【参考答案】

由初始数组建立小根堆的过程如下：



基于小根堆，进行堆排序的过程如下：



18. (8 分) 假设已经有 k 个长度分别为 M_0, M_1, \dots, M_{k-1} 的有序表, 现通过两两合并的方式将它们合并为一个有序表, 若要使合并过程中元素的总比较次数最小, 应该按照什么次序进行合并? 说明你的理由, 必要时可以举例说明。

考点: 排序算法的应用 (归并排序)

来源: 四川大学874 2019年

【参考答案】

长为 m 和 n 的两个有序表合并，最坏情况下，关键字比较次数为 $m+n-1$ 次。最坏情况下关键字对比次数依赖于两表的表长之和。因此每次合并时，应挑选表长最短的两个有序表进行合并，直到最终合并为一个表为止。类似哈夫曼树的思想。

咸鱼注：根据题目的要求，你可以自己举一个例子，比如，设定有5个有序表，表长分别是？，？，？，？，？，然后画出归并树。此处不再举例，聪明的你一定可以搞定的，对吗？

咸鱼注：打字画图不易，来动动小手关注一波微博哈，祝各位咸鱼翻身！ @王道咸鱼老师-计算机考研



接头暗号：

