

DSA2016 真题 + 解答

一、判断(2'*20)

1. 若AVL树插入元素的过程中发生了旋转操作，则树高必不变。
– **正确。**（07-D3、P755-756）AVL树失衡后通过调整必回复原高度（如果没出现失衡，则树高可能发生改变）。
2. 若红黑树插入一个元素后黑高度增加，则双红修正过程中没有拓扑结构变换，只有重染色操作。
– **正确。**（08-C3、P841-852）当黑高度增高时，只能是一直发生双红修正的RR-2情况，此情况下不会发生“3+4”重构（即拓扑结构变换），只会发生重染色（每次染3个，最多染 $O(\log n)$ 个）。
3. 若KMP算法不使用改进版的next表，最坏情况下时间复杂度可能达到 $O(mn)$ 。
– **错误。**最坏情况下KMP的实践复杂度为 $O(m+n)$ 。
4. 在BST中删除两个节点（7-B3），则无论先删除哪个节点，最终BST的拓扑结构均相同。
– **正确。**
5. 完全二叉堆删除元素在最坏情况下时间复杂度为 $O(\log n)$ ，但平均情况下仅为 $O(1)$ 。
– **错误。**删除元素要用下滤操作，而下滤操作在平均情况下时间复杂度也是 $O(\log n)$ 的。
6. 在任何情况下，伸展树总能保持每次操作 $O(\log n)$ 的平均复杂度。
– **错误。**伸展树的均摊复杂度为 $O(\log n)$ ；但像题目这样说是对其的吗？我偏向错误。
7. 对于左式堆A和B，合并后所得二叉堆的右侧链元素一定来自A和B的右侧链。
– **错误。**（10-XB2，P1211-1219）由于可能会交换左右子堆，所以不一定保证（而且书上的例子就是一个反例）。
8. 如果元素理想随机，那么对二叉搜索树做平衡化处理，对改进其渐进时间复杂度没有什么作用。
– **疑问。**元素理想随机的话，树高期望是 $\log n$ 量级还是 \sqrt{N} 量级的呢？
9. 采用双向平方试探策略时，将散列表长度取作素数 $M=4k+3$ ，可以极大地降低查找前M个位置冲突的概率，但仍不能杜绝。
– **错误。**（09-C5，P1058，习题解析9-17）能够杜绝。
10. 在使用Heapify批量建堆的过程中，改变同层节点的下滤次序对算法的正确性和时间数常数无影响。
– **正确。**同层节点下滤的互不相干的。

11. 在kd-search中, 查找区间R与任一节点的4个孙节点(假设存在)对应区域最多有两个相交。
 - 正确。(08-F6, P914-918, 习题解析8-16) 书上的结论写的是四个孙节点中的至多两个节点会相交(注意这里对相交的定义)。
12. 在n个节点的跳转表中, 塔高的平均值为 $O(\log n)$ 。
 - 错误。期望的塔高为2。
13. 既然可以在 $O(n)$ 时间内找出n个数的中位数, 快速排序算法(12-A1)即可优化至 $O(n \log n)$ 。
 - 疑问。我偏向错误。快速排序本来就是(均摊? 分摊? 平均?) $O(n \log n)$ 的; 找出中位数本身对时间是得不偿失的。
14. 将N个关键码按随机次序插入B树, 则期望的分裂次数为 $O(\log^2 N)$ 。
 - 正确。(08-B4, P811-821) 在树高为h的情况下, 想要整体增加1, 则肯定需要涉及到连续h次分裂操作(上溢), 所以高度为h的B树期望的分裂次数为 $O(h^2)$ 。
15. 与二叉堆相比, 多叉堆delMax()操作时间复杂度更高。
 - 错误。(10-XA1, 1201) 这里说的是下滤的成本会更高, 但一般计算的话三叉堆是比二叉堆要快的。
16. 若元素理想随机, 则用除余法作为散列函数时, 即使区间长度不是素数, 也不会影响数据的均匀性。
 - 正确。
17. 与胜者树相比, 败者树在重赛过程中需反复将节点与其兄弟进行比较。
 - 错误。胜者树才需要在重赛过程中反复将节点与其兄弟进行比较。
18. 在图的优先级搜索中, 每次可能调用多次prioUpdater, 但累计调用次数仍为 $O(e)$ 。
 - 正确。
19. 若序列中逆序对个数为 $O(n^2)$, 则使用快速排序(12-A1)进行的交换次数为 $O(n \log n)$ 。
 - 疑问。。通过构造可以使交换次数达到 $O(n)$ 。
20. 如果把朋友圈视为一无向图, 那么即使A君看不到你给B君点的赞, 你们仍可能属于同一个双连通分量。
 - 正确。

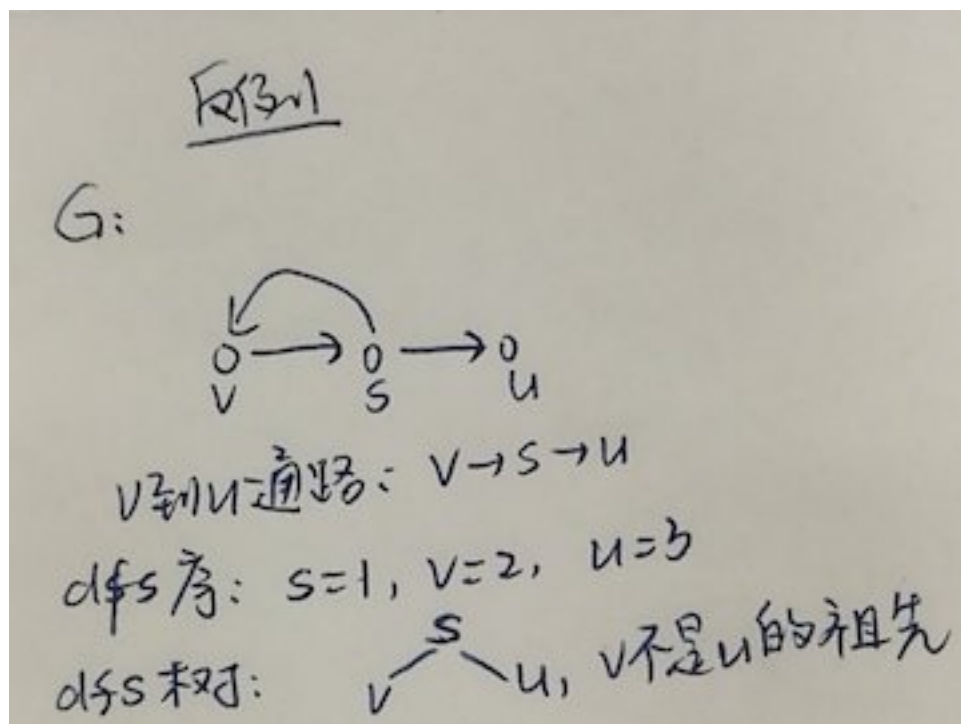
二、选择(3' * 10)

1. 二叉堆中某个节点秩为K, 则其兄弟节点(假设存在)秩为 (D)
 - A. $K+1$ B. $K-1$ C. $K+(-1)^K$ D. $K-(-1)^K$ E. 以上皆非
 - 注: 试一下就好了; 但要注意, 根节点秩为0。
2. 由5个互异节点构成的不同的BST共有 (D)

- A. 24 B. 30 C. 36 **D. 42** E. 120
 - 注：卡特兰数
3. 有2015个节点的左式堆，左子堆最小规模为 (E)
 - A. 10 B. 11 C. 1007 D. 1008 E. 以上皆非
 - 注：1个
 4. 与MAD相比，除余法在 (BC) 有缺陷
 - A. 计算速度 **B. 高阶均匀性 C. 不动点** D. 满射性 E. 以上皆非
 5. 以下数据结构在插入元素后可能导致 $O(\log n)$ 次局部结构调整的是 (BD)
 - A. AVL **B. B树** C. 红黑树 **D. 伸展树** E. 以上皆非
 - 注：B树-连续上溢；伸展树-旋转到根；红黑树-不知道染色算不算局部结构调整
 6. 对小写字母集的串匹配，KMP算法与蛮力算法在 (AC) 情况下渐进时间复杂度相同
 - **A. 最好** B. 最坏 **C. 平均** D. 以上皆非
 7. 对随机生成的二进制串，gs表中 $gs[0] = 1$ 的概率为 ()
 - A. $1/2^m$ **B. $1/2^{(m-1)}$** C. $1/2^{(m+1)}$ D. $1/m$
 - 注：所有元素相等
 8. 以下数据结构，空间复杂度为线性的是 (ACD)
 - **A. 2d tree** B. Range tree **C. Interval tree D. Segment tree** E. 以上皆非
 9. 人类拥有的数字化数据总量，在2010年已经达到2B ($2^{20} = 10^{21}$) 量级，若每个字节自成一个关键码，用一棵16阶B-树存放，则可能的高度为 (B)
 - A. 10 **B. 20** C. 40 D. 80 E. >80
 10. 在BST中查找时候，以下查找序列不可能出现的是 (BD) 【选项是自己大概编的】
 - A. 912, 209, 911, 265, 344, 380, 365
 - **B. 87, 768, 456, 372, 326, 378, 365**
 - C. 48, 260, 570, 307, 340, 380, 361, 365
 - **D. 726, 521, 201, 328, 384, 319, 365**

三、综合 (10'*3)

1. 在有向图G中，存在一条自顶点v通向u的路径，且在某次DFS中有 $dTime(v) < dTime(u)$ ，则在这次DFS所生成的DFS森林中，v是否一定是u的祖先？若是，请给出证明，若不是，请举出反例。



2. 对闭散列 $[0, M)$, $M=2^s$, 采用如下冲突排解方法: 初始时 $c=0$, $d=0$, 探查key冲突时, $c \rightarrow c+1$, $d \rightarrow d+c$, 探查 $H[(key+d) \% M]$ 。则这种算法是否可以保证空间能被100%利用? 若是, 请给出证明, 若不是, 请举出反例。

设 $f(d) = \frac{1}{2}d(d+1)$. 则需要证明:
 $\forall p \in [0, m), \exists d_0 \text{ s.t. } f(d_0) \% M \equiv p.$

以下反证.

$$\textcircled{1} f(d) \equiv f(d+M) \pmod{M}.$$

$$\begin{aligned} & f(d+M) \pmod{M} \\ &= \frac{1}{2}(d+M)(d+1+M) \pmod{M} \\ &= \frac{1}{2}d(d+1) + \frac{M}{2}d(d+1) + \frac{M^2}{2} \pmod{M} \\ &= f(d) + \underbrace{\frac{M}{2}d(d+1)}_{\equiv KM} + \underbrace{\frac{M^2}{2}}_{\substack{\equiv LM \\ M=2S}} \pmod{M} \\ &= f(d) \pmod{M}. \end{aligned}$$

故只需证:

$$\forall p \in [0, m), \exists d_0 \in [0, M), \text{ s.t. } f(d_0) \% M \equiv p.$$

$\textcircled{2}$ 若 $\exists p \in [0, m)$. 没有 d_0 与之对应.

则根据鸽巢原理, 必有 $\exists p_0, \exists i, j \in [0, M)$.

$$\text{s.t. } f(i) \equiv f(j) \pmod{M} \\ \equiv p_0.$$

$$\text{故有 } f(i) - f(j) \equiv 0 \pmod{M}$$

$$\text{即 } \frac{1}{2} i(i+1) - \frac{1}{2} j(j+1) \equiv 0 \pmod{M}$$

$$i^2 - j^2 + i - j \equiv 0 \pmod{2M}$$

$$(i+j+1)(i-j) \equiv 0 \pmod{2M}.$$

$$\text{又 } i \neq j, \text{ 故 } i-j \neq 0$$

$$i, j \leq M, \text{ 故 } i+j+1 \neq 2M$$

又 $i+j+1$ 与 $i-j$ 必定一奇一偶, $2M$ 为偶
~~和~~ 余数不可能为 0. 故矛盾.

故得证。

3. 在不改变BST和BinNode定义的前提下（BinNode仅存储着parent, data, rc, lc）。设计算法，使得从节点x出发，查找值为Y的节点y，时间复杂度为 $O(d)$ ，d为节点x与y的距离，要求利用树的局部性，复杂度与总树高无关，否则将不能按满分起评。

(函数定义域，参量为BinNode x,T,Y，返回值为BinNode类型，函数名fingerSearch（不会写成代码orz...）)

- (a) 说明算法思路
- (b) 写出伪代码
- (c) 在图中画出值为6的点查找值为17的点的查找路径
- (d) 说明算法时间复杂度为 $O(d)$ （若无法达到，说明困难在哪）