1、Insert 2, 1, 4, 5, 9, 3, 6, 7 into an initially empty AVL tree. Which one of the following statements is FALSE?

A.4 is the root

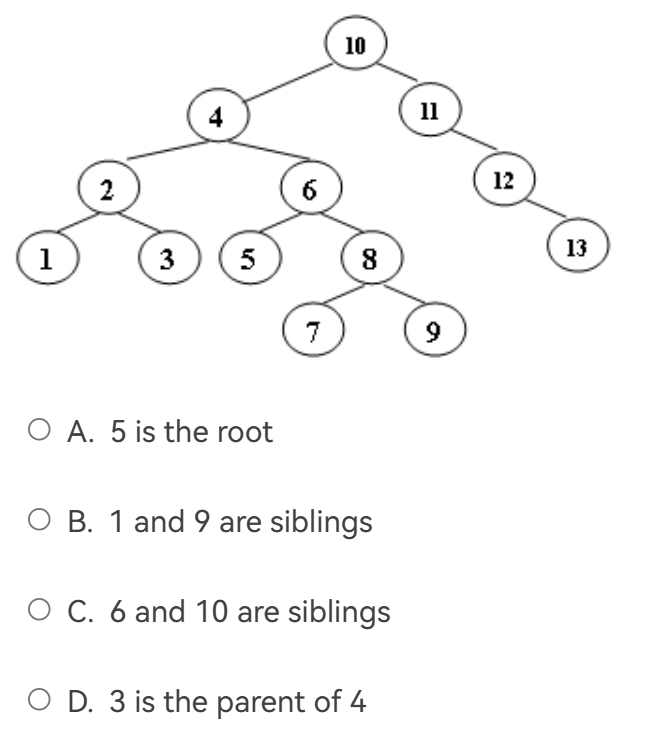
B.3 and 7 are siblings

C.2 and 6 are siblings

D.9 is the parent of 7

B

1. For the result of accessing the keys 3, 9, 1, 5 in order in the splay tree in the following figure, which one of the following statements is FALSE?



[AVL树](https://so.csdn.net/so/search?q=AVL%E6%A0%91&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/Woolseyyy/article/details/_blank)通过使树balanced降低树的高度至logN，从而使得单次搜索的复杂度为O(logN)。而Splay Trees则通过每次将被访问到的节点置于根，使得m次连续搜索的复杂度为O(mlogN)。

D 一个一个画

1. When doing amortized analysis, which one of the following statements is FALSE?

A.Aggregate analysis shows that for all n, a sequence of n operations takes worst-case time T(n) in total. Then the amortized cost per operation is therefore T(n)/n

B.For potential method, a good potential function should always assume its maximum at the start of the sequence

C.For accounting method, when an operation's amortized cost exceeds its actual cost, we save the difference as credit to pay for later operations whose amortized cost is less than their actual cost

D.The difference between aggregate analysis and accounting method is that the later one assumes that the amortized costs of the operations may differ from each other

B

4、 Consider the following buffer management problem. Initially the buffer size (the number of blocks) is one. Each block can accommodate exactly one item. As soon as a new item arrives, check if there is an available block. If yes, put the item into the block, induced a cost of one. Otherwise, the buffer size is doubled, and then the item is able to put into. Moreover, the old items have to be moved into the new buffer so it costs k+1 to make this insertion, where k is the number of old items. Clearly, if there are N items, the worst-case cost for one insertion can be Ω(N). To show that the average cost is O(1), let us turn to the amortized analysis. To simplify the problem, assume that the buffer is full after all the N items are placed. Which of the following potential functions works?

A.The number of items currently in the buffer

B.The opposite number of items currently in the buffer

C.The number of available blocks currently in the buffer

D.The opposite number of available blocks in the buffer

D

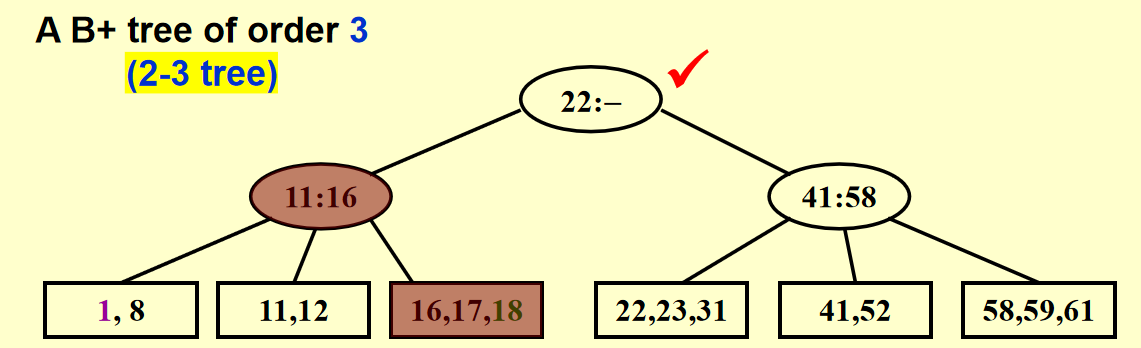
选项A（当前缓冲区中的项目数量）和B（当前缓冲区中项目数量的相反数）都不能很好地反映每次插入操作的实际成本。因为在每次插入操作时，项目数量并不总是准确反映出实际的开销。当缓冲区有可用块时，插入成本为1，而不考虑当前的项目数量。

选项C（当前缓冲区中的可用块数量）是一个不错的选择，因为插入操作的成本与可用块数量相关。然而，选项D（可用块数量的相反数）更具优势，因为它能够更准确地反映出插入操作的实际成本变化。当可用块不足时，插入操作的成本为负数，表示需要进行扩展操作。每次插入新项目后，潜在函数的值会递增，增量为1。当扩展操作发生时，潜在函数的值会减少，减量为旧项目的数量加一。这样，选项D能够更准确地反映出每次插入操作的实际成本变化，从而更好地支持摊销分析。

5、A 2-3 tree with 3 nonleaf nodes must have 18 keys at most.

一个叶子节点最多放3个值，然后是两个节点指向3个叶子，然后是一个根指向两个节点。

那么一共有2\*3个叶子，一个叶子存放3个key，一共18个.注意key只算叶子上的。T



6、In the red-black tree that results after successively inserting the keys 41; 38; 31; 12; 19; 8 into an initially empty red-black tree, which one of the following statements is FALSE?

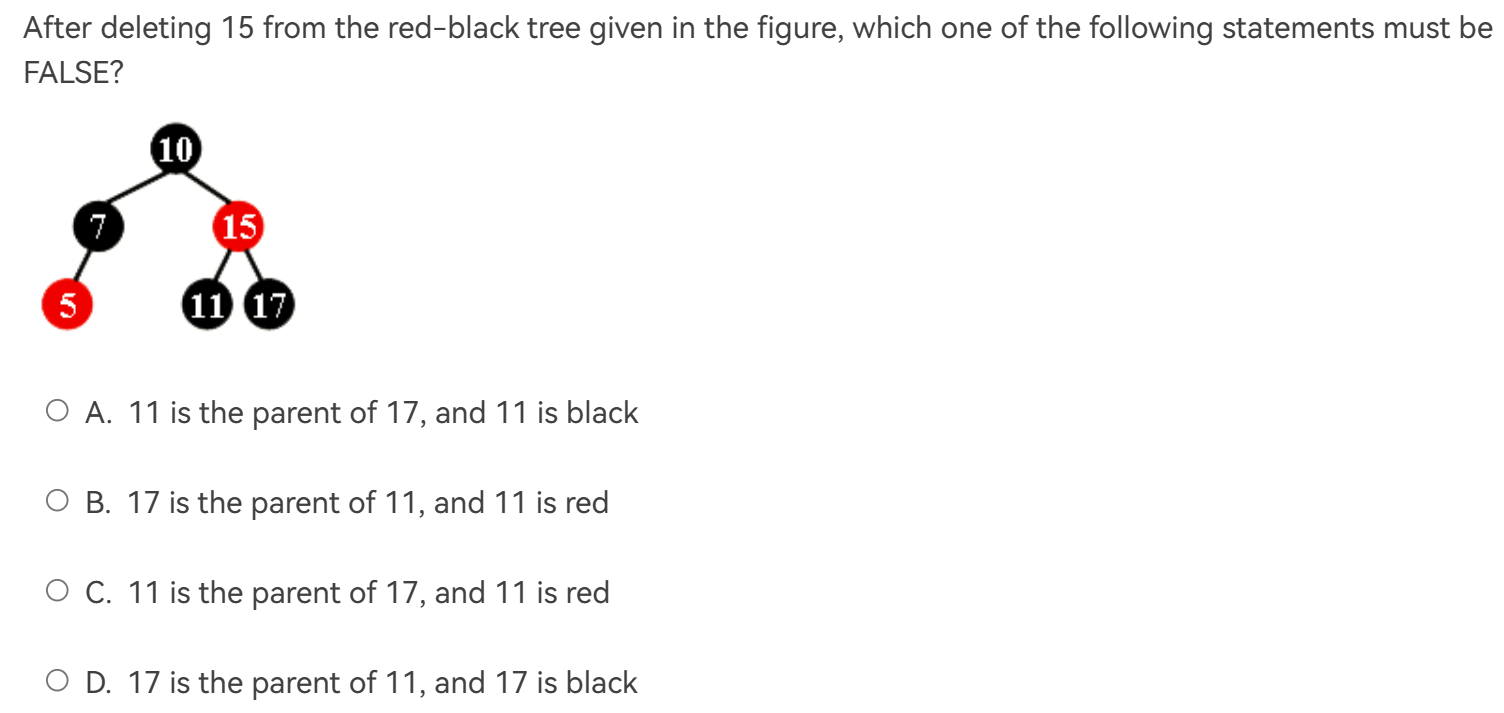
A.38 is the root

B.19 and 41 are siblings, and they are both red

C.12 and 31 are siblings, and they are both black

D.8 is red

B 按照case一层一层调整

7、



11

17

15

11

17

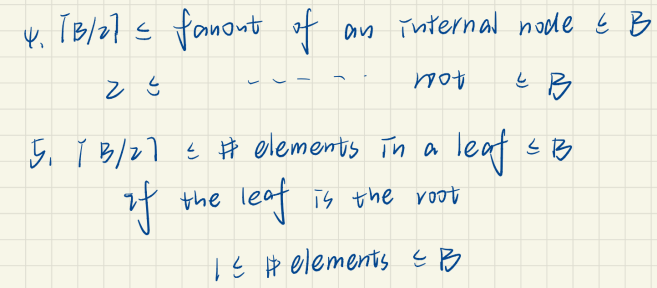
15

11

17

15

C 左子树的最大值（11）和要删除的15交换，那么15变成双黑。如果11是红色那么从11这个节点出发的路径不满足红黑树的第5条性质

8、Insert 3, 1, 4, 5, 9, 2, 6, 8, 7, 0 into an initially empty 2-3 tree (with splitting). Which one of the following statements is FALSE?

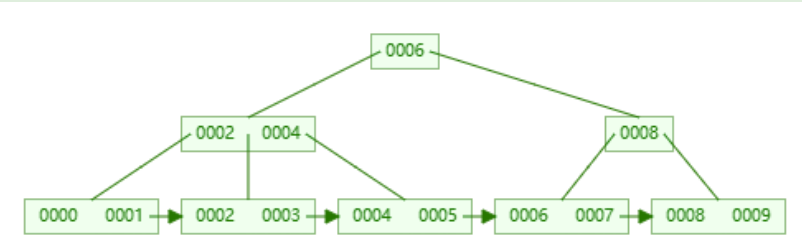
A.7 and 8 are in the same node

B.the parent of the node containing 5 has 3 children

C.the first key stored in the root is 6

D.there are 5 leaf nodes

A



9、Which of the following statements concerning a B+ tree of order M is TRUE?

A.the root always has between 2 and M children

B.not all leaves are at the same depth

C.leaves and nonleaf nodes have some key values in common

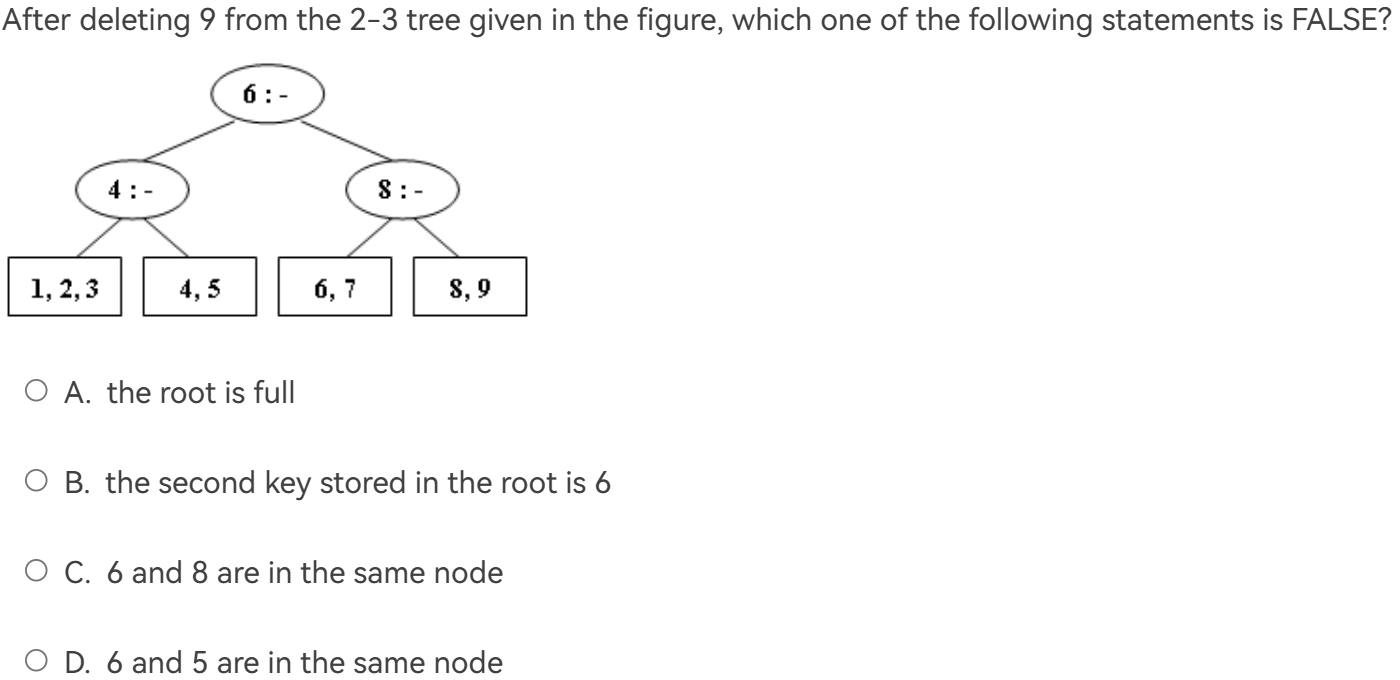
D.all nonleaf nodes have between ⌈M/2⌉ and M children

A:当root是叶子时错——0个孩子

B:错all leaves are at the same depth

C:对，某些key是相同的，因为key值只算叶子结点，其他算索引。

D:(M-1)/2 to M-1 个key, M/2到M个children, 但是根可以只有两个。（即根结点至少拥有两颗子树。除了根结点以外，其余每个分支结点至少拥有M/2[课子树](https://www.zhihu.com/search?q=%E8%AF%BE%E5%AD%90%E6%A0%91&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":3219695465}" \t "https://www.zhihu.com/question/_blank)）

10、After deleting 9 from the 2-3 tree given in the figure, which one of the following statements is FALSE?

6,7,8

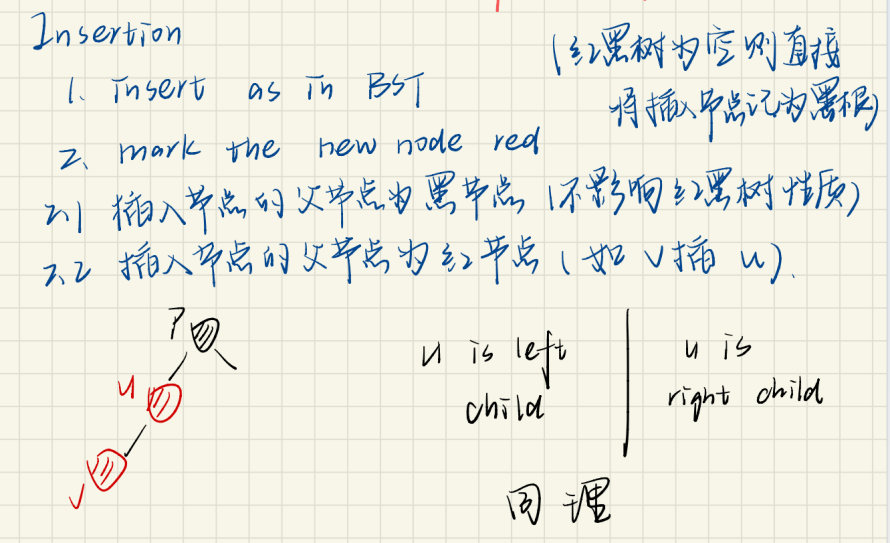
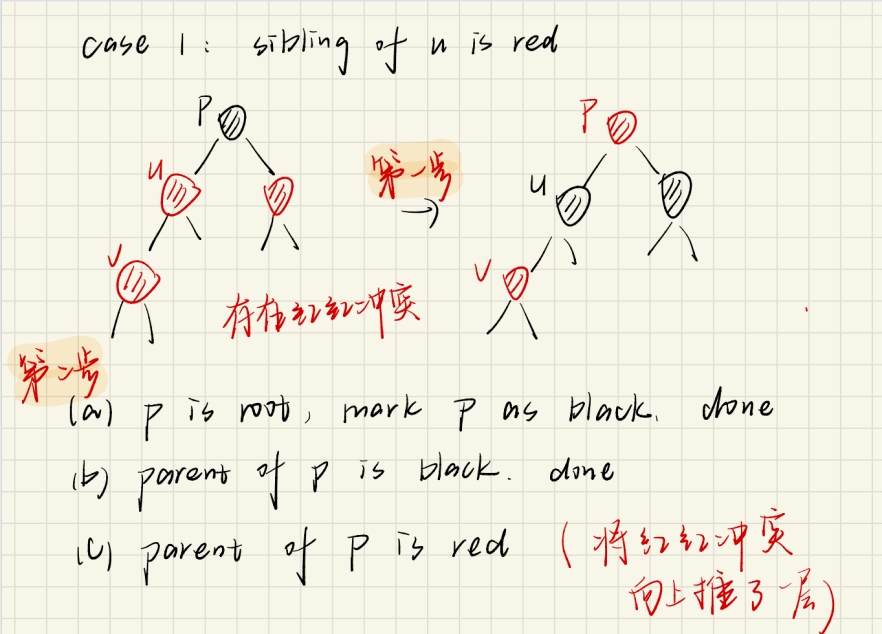
4,5

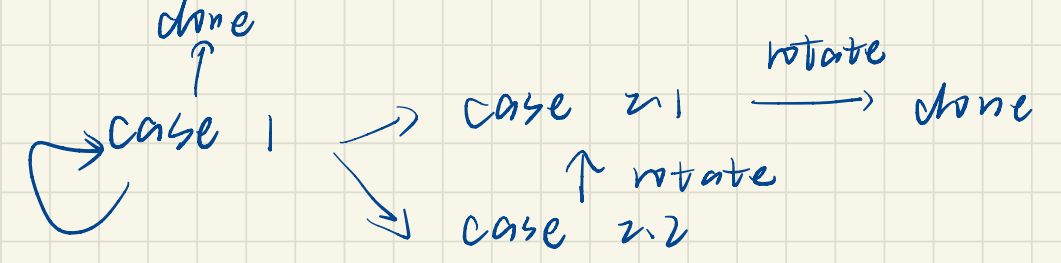
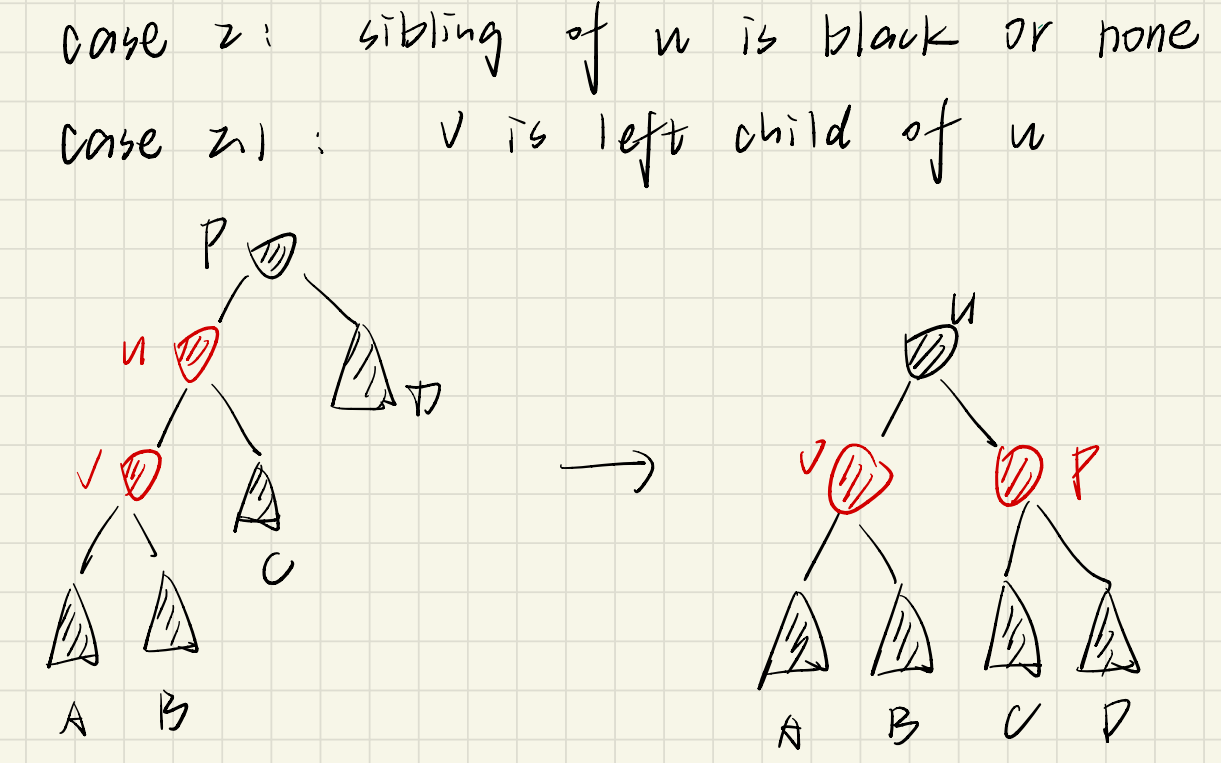
1,2,3

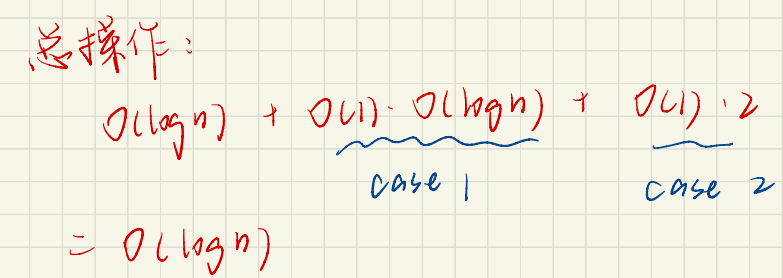
4:6

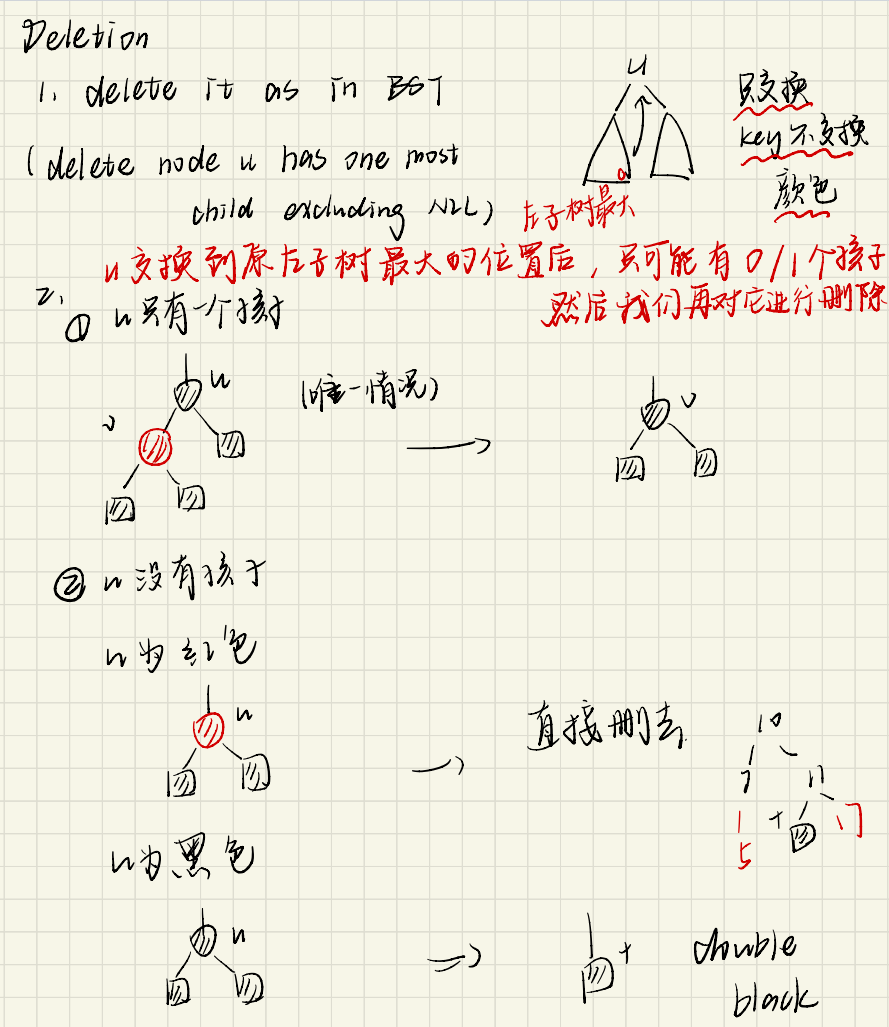
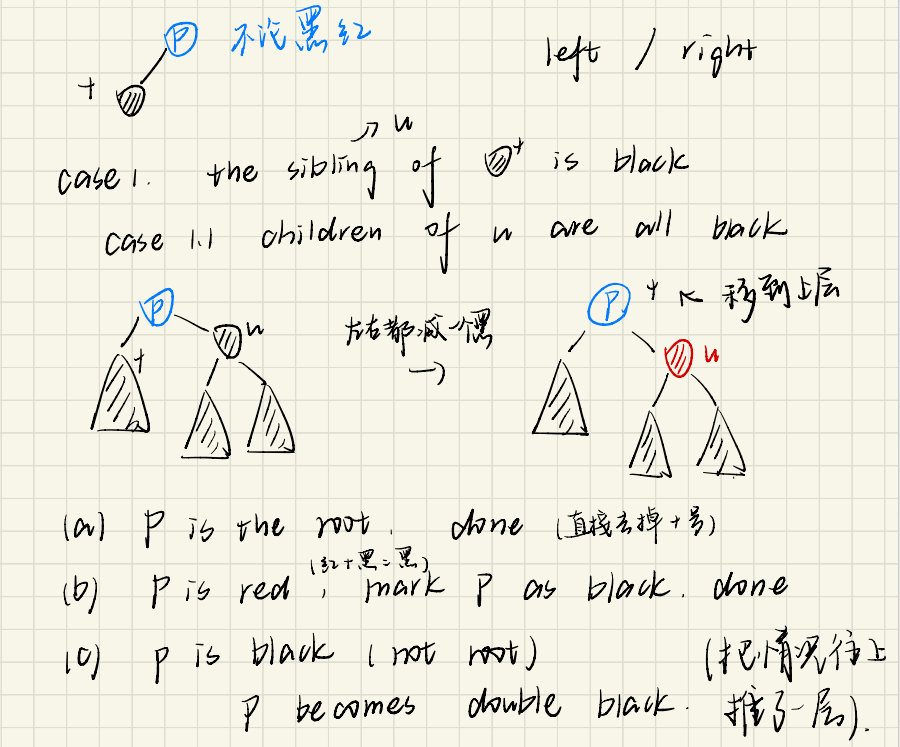
1. 删除9之后由于太少了，6,7,8会合并，进而向上合并

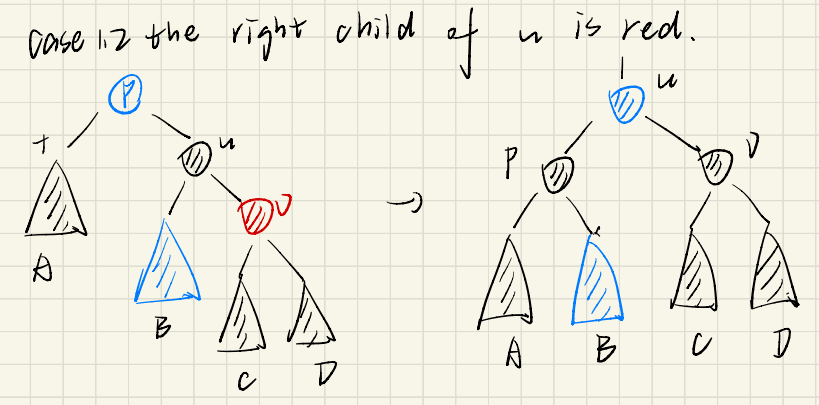
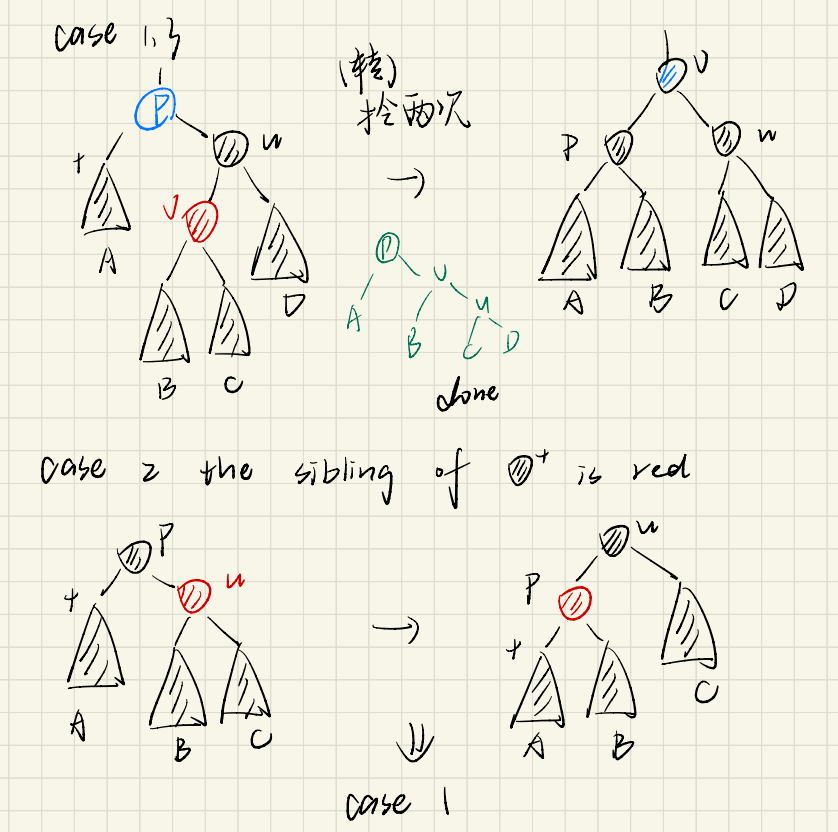
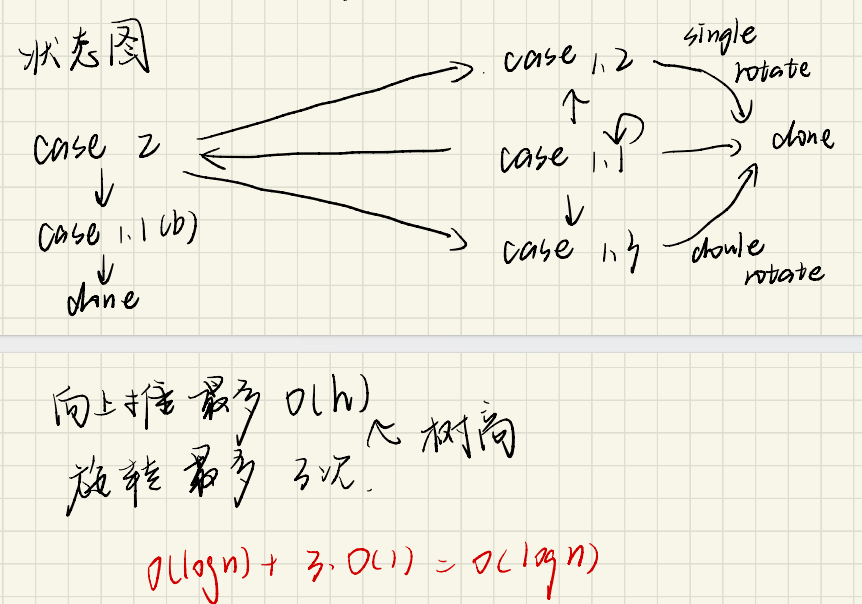
红黑树插入删除











近侄子节点

远侄子节点

1. In distributed indexing, document-partitioned strategy is to store on each node all the documents that contain the terms in a certain range.

译：在分布式索引中，文档分区策略是在每个节点上存储包含特定范围内术语的所有文档。

答：[磁盘分区](https://so.csdn.net/so/search?q=%E7%A3%81%E7%9B%98%E5%88%86%E5%8C%BA&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/HGGshiwo/article/details/_blank)索引技术，每个节点存放部分索引，不是所有的。F

1. When evaluating the performance of data retrieval, it is important to measure the relevancy of the answer set.

译：在评估数据检索的性能时，衡量答案集的相关性非常重要。

Data Retrieval Performance Evaluation考虑Response time和Index space

Information Retrieval Performance Evaluation考虑relevancy

F

13、Precision is more important than recall when evaluating the explosive detection in airport security.

机场安全，应该是返回的全面度更重要，精确性不算重要。F

14、When measuring the relevancy of the answer set, if the precision is high but the recall is low, it means that:

A.most of the relevant documents are retrieved, but too many irrelevant documents are returned as well

B.most of the retrieved documents are relevant, but still a lot of relevant documents are missed

C.most of the relevant documents are retrieved, but the benchmark set is not large enough

D.most of the retrieved documents are relevant, but the benchmark set is not large enough

B

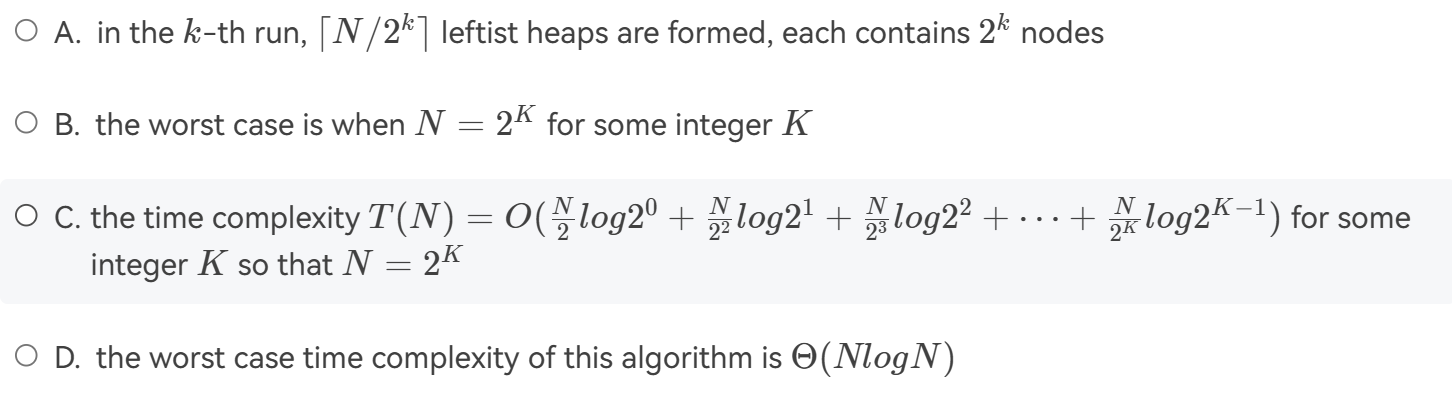
1. The result of inserting keys 1 to 2k−1 for any k>4 in order into an initially empty skew heap is always a full binary tree.

T

1. The right path of a skew heap can be arbitrarily long.

T

1. We can perform BuildHeap for leftist heaps by considering each element as a one-node leftist heap, placing all these heaps on a queue, and performing the following step: Until only one heap is on the queue, dequeue two heaps, merge them, and enqueue the result. Which one of the following statements is FALSE?



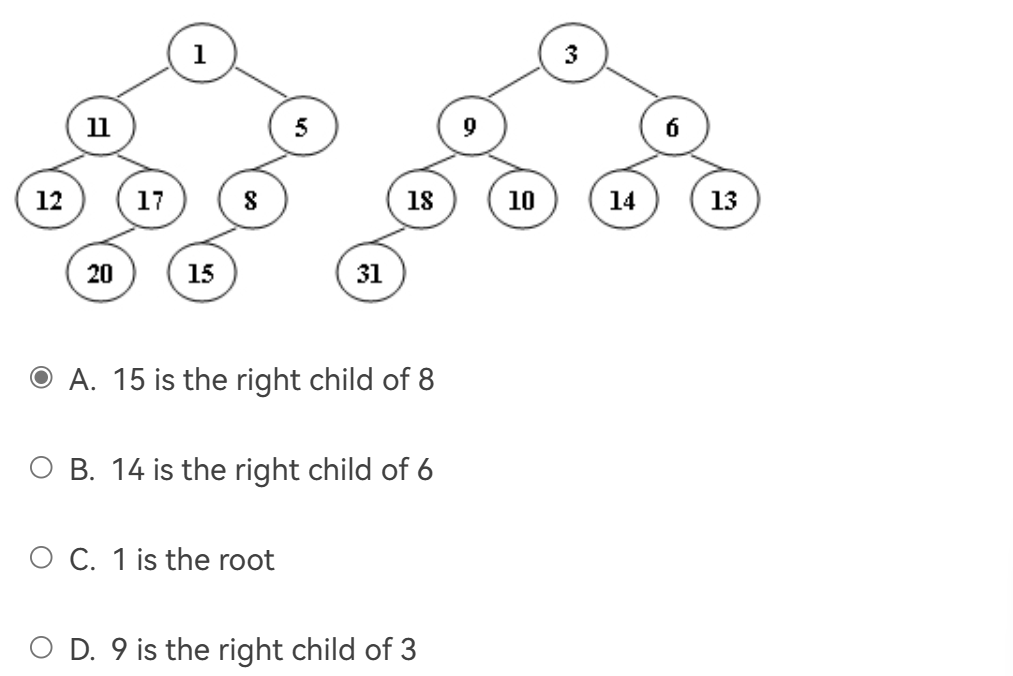
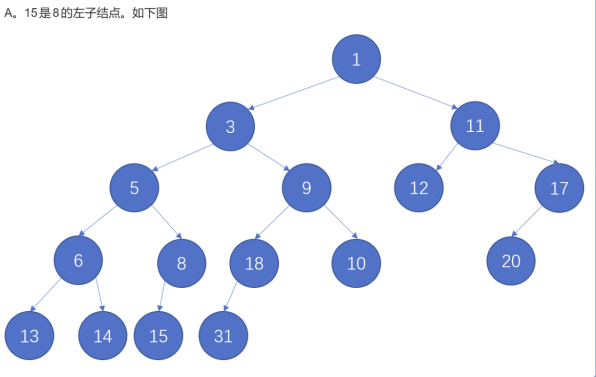
选D.

A.第 k 次合并中，我们从队列中取出两个堆进行合并，因此每次合并会减少队列中的堆数量一半。因此，在第 k 次合并后，会形成 ⌈N/2^k⌉ 个左倾堆，每个左倾堆包含 2^k 个节点

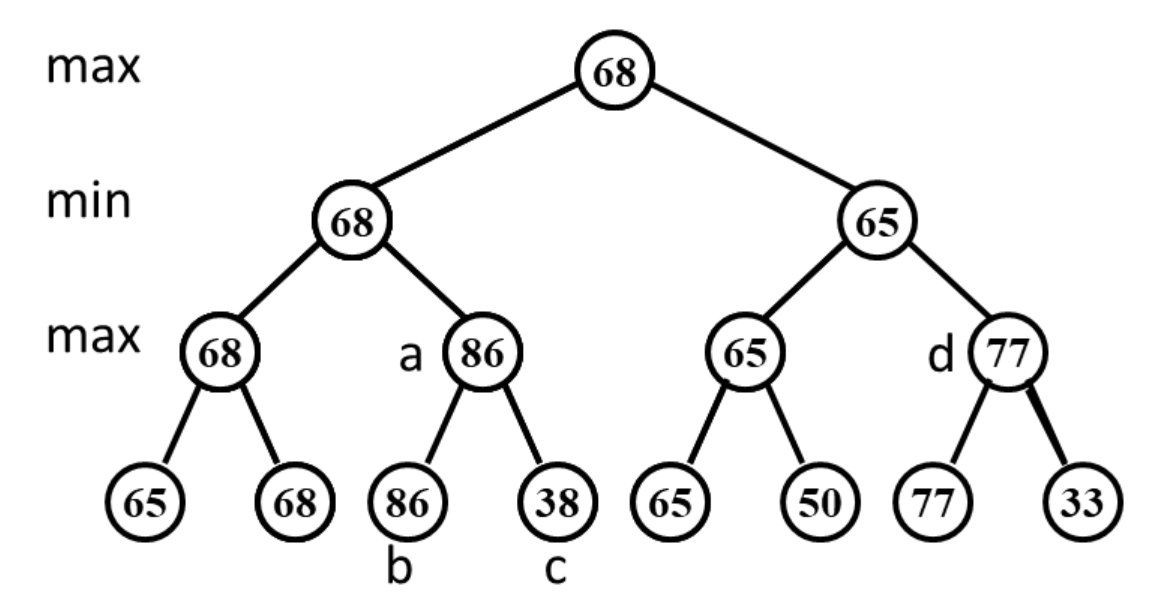
B.在最坏情况下，堆的数量减少得非常缓慢，每次合并都只能减少一个堆。如果最终只剩下一个堆之前，每次合并都能减少多个堆，那么左倾堆的构建过程将会更快。但是，左倾堆的性质决定了每次合并只能合并两个堆，因此在最坏情况下，左倾堆构建完成时只剩下一个堆是合理的。

C.分治，和归并差不多。T(N)=2T(N/2)+O(logN),复杂度为O(N)

18、Merge the two skew heaps in the following figure. Which one of the following statements is FALSE?



19、Given the following game tree, which node is the first one to be pruned with α-β pruning algorithm?

C

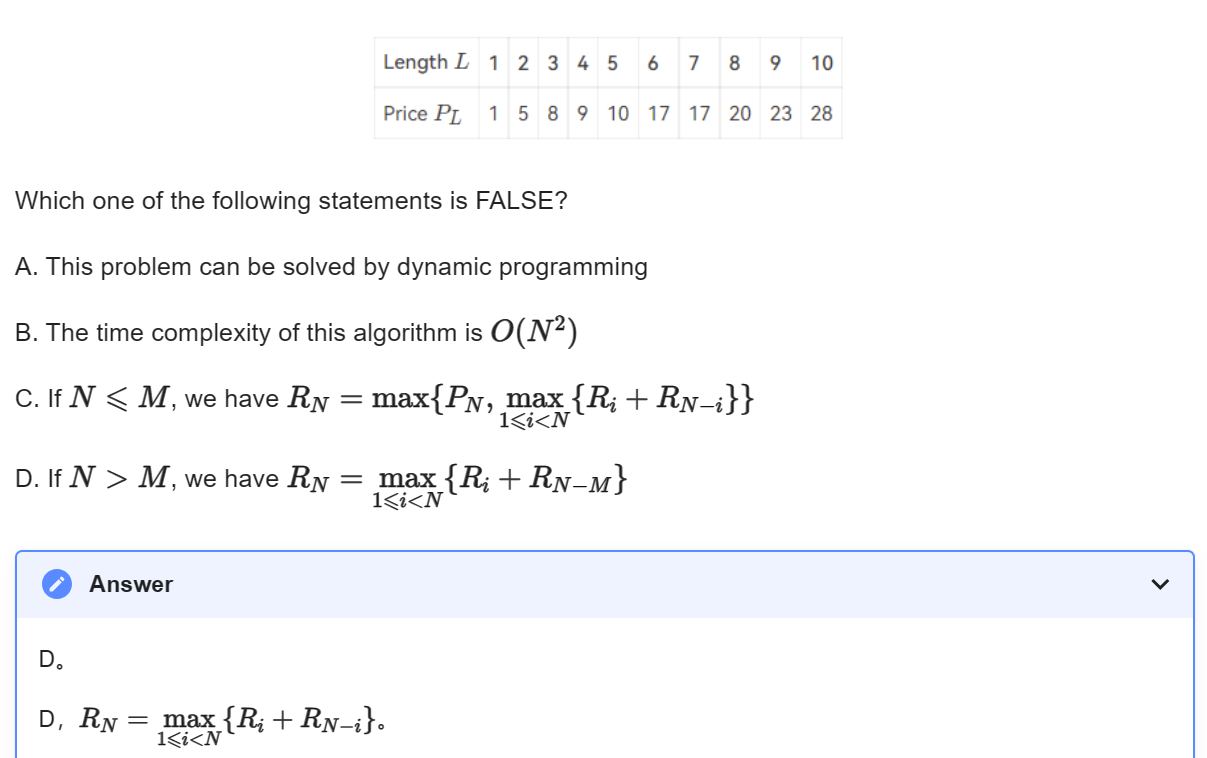
20、Which one of the following is the lowest upper bound of T(n) for the following recursion T(n)=2T(√n)+logn?  
A. O(lognloglogn)

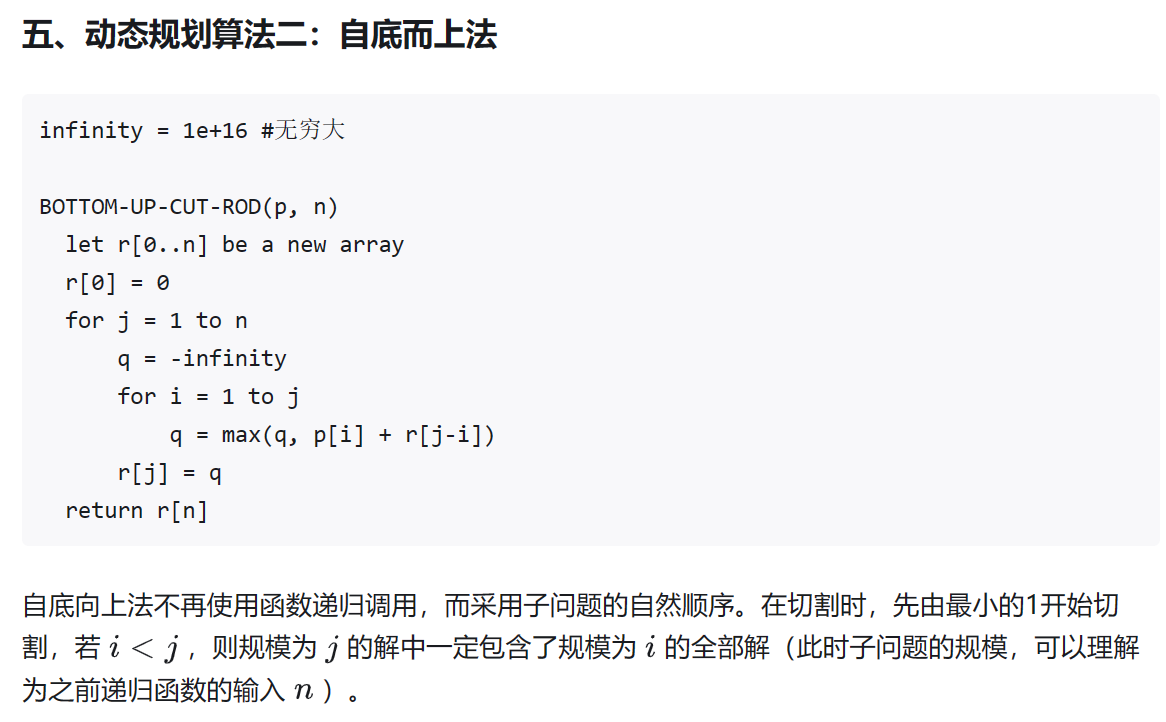
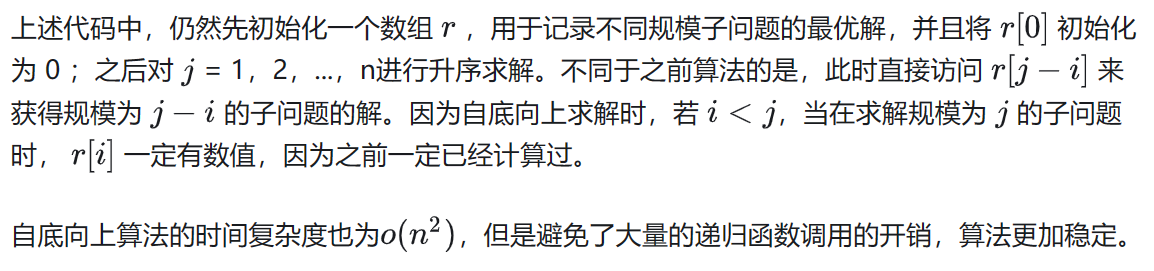
B. O(log2n)

C. O(nlogn)  
D. O(n2)

答：

设 m=logn, 则2m=n.  
T(2m)=2T(2m/2)+m  
设 G(m)=T(2m)，则原式转化为G(m)=2G(m/2)+m  
根据主定理，a=2,b=2,k=1,p=0,a=bk,满足条件2，所以算法复杂度为O（mlogm）又因为 m=logn,所以算法复杂度为O（lognloglogn)

1. Q2-1. Rod-cutting Problem: Given a rod of total length N inches and a table of selling prices PL for lengths L=1,2,⋯,M. You are asked to find the maximum revenue RN obtainable by cutting up the rod and selling the pieces. For example, based on the following table of prices, if we are to sell an 8-inch rod, the optimal solution is to cut it into two pieces of lengths 2 and 6, which produces revenue R8=P2+P6=5+17=22. And if we are to sell a 3-inch rod, the best way is not to cut it at all.



1. Let S be the set of activities in Activity Selection Problem. Then the earliest finish activity am must be included in all the maximum-size subset of mutually compatible activities of S.

F

最优解不一定是贪婪最优解。

1. Let C be an alphabet in which each character c in C has frequency c.freq. If the size of C is n, the length of the optimal prefix code for any character c is not greater than n−1.

T 霍夫曼树

24、All the languages can be decided by a non-deterministic machine.  
F  
不确定[图灵机](https://so.csdn.net/so/search?q=%E5%9B%BE%E7%81%B5%E6%9C%BA&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/HGGshiwo/article/details/_blank)可以用来验证NP问题的解是否是正确的，确定图灵机可以用来求解P问题。

NP hard问题无法通过不确定图灵机验证

25、All NP problems can be solved in polynomial time in a non-deterministic machine.

T

26、Among the following problems, \_\_ is NOT an NP-complete problem.

A.Vertex cover problem

B.Hamiltonian cycle problem、Traveling Salesman Problem

C.Halting problem

D.Satisfiability problem

D SAT问题是第一个被证明的NPC问题，A是NPC问题，B是汉密尔顿回路，NPC问题。C停机问题是不可解的，选C

27、Suppose Q is a problem in NP, but not necessarily NP-complete. Which of the following is FALSE?

A.A polynomial-time algorithm for SAT would sufficiently imply a polynomial-time algorithm for Q.

B.A polynomial-time algorithm for Q would sufficiently imply a polynomial-time algorithm for SAT.

C.If Q ∉P, then P≠NP.

D.If Q is NP-hard, then Q is NP-complete.

SAT是NPC问题，如果解决了，可以解决所有NP问题

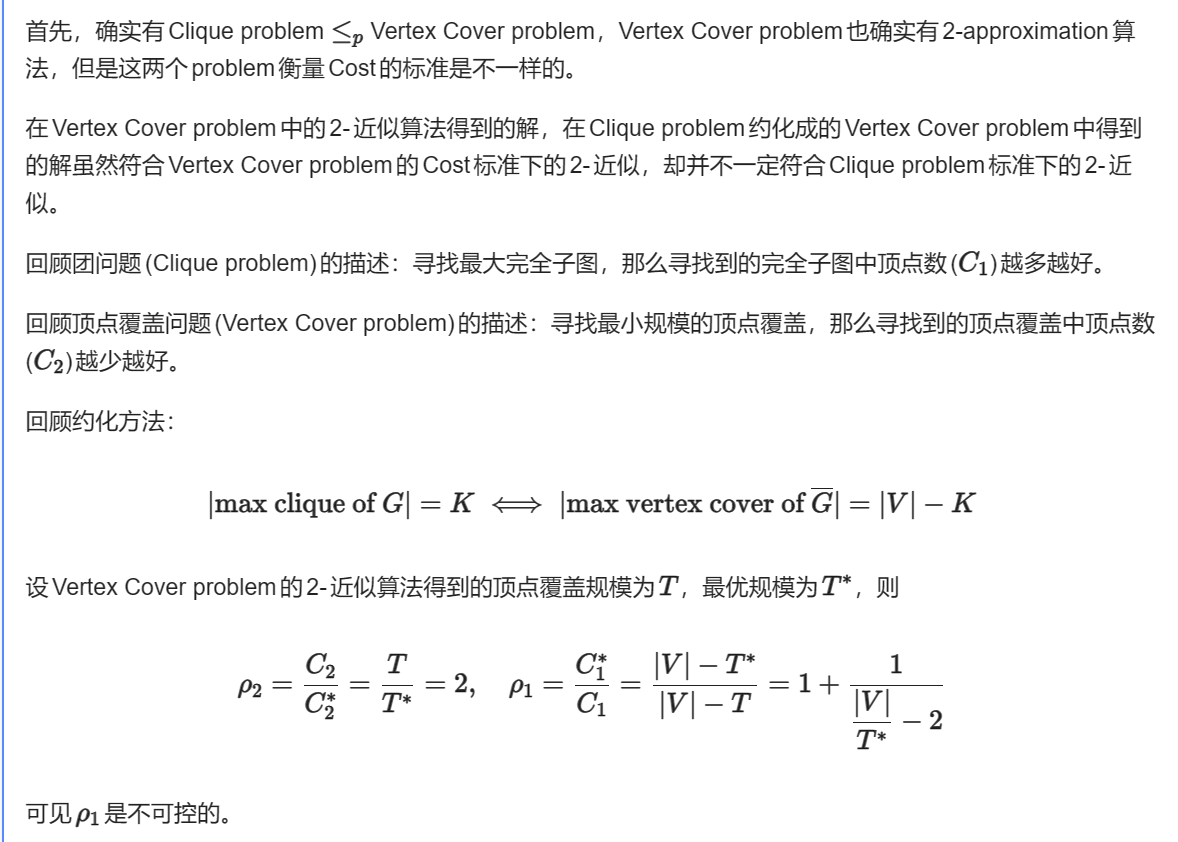
B：Q不一定是NPC的，所以不对；C：如果Q不是P，那么说明NP没有被解决；D：NP-hard和NP交集是NPC

1. Suppose ALG is an α-approximation algorithm for an optimization problem ∏∏ whose approximation ratio is tight. Then for everyε>0 there is no (α−ε)-approximation algorithm for ∏ unless P = NP.

(F)对于一种算法而言，近似比为 α ，那么∀β>α ，都可以说 β 是其近似比。如果 α 是 tight 的，则 α 是一个下确界。但这都只是对这一种算法的分析，一个 tight 的近似比只能说明你对这种算法的分析到位了，而不能说明这个问题没有更好的算法。这里完全是两码事

1. As we know there is a 2-approximation algorithm for the Vertex Cover problem. Then we must be able to obtain a 2-approximation algorithm for the Clique problem, since the Clique problem can be polynomially reduced to the Vertex Cover problem.

(F)

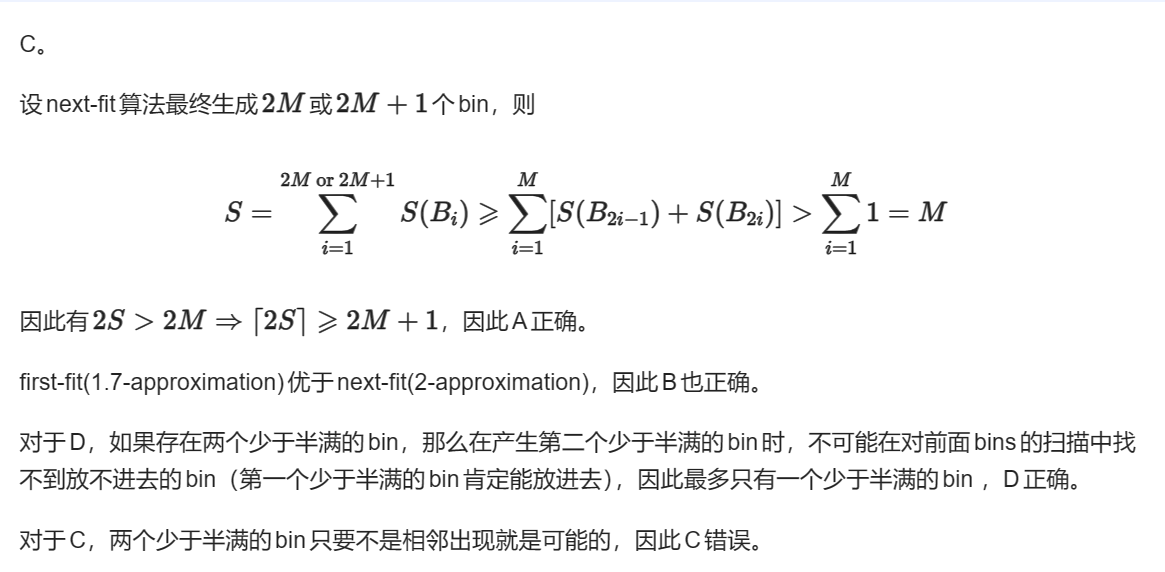


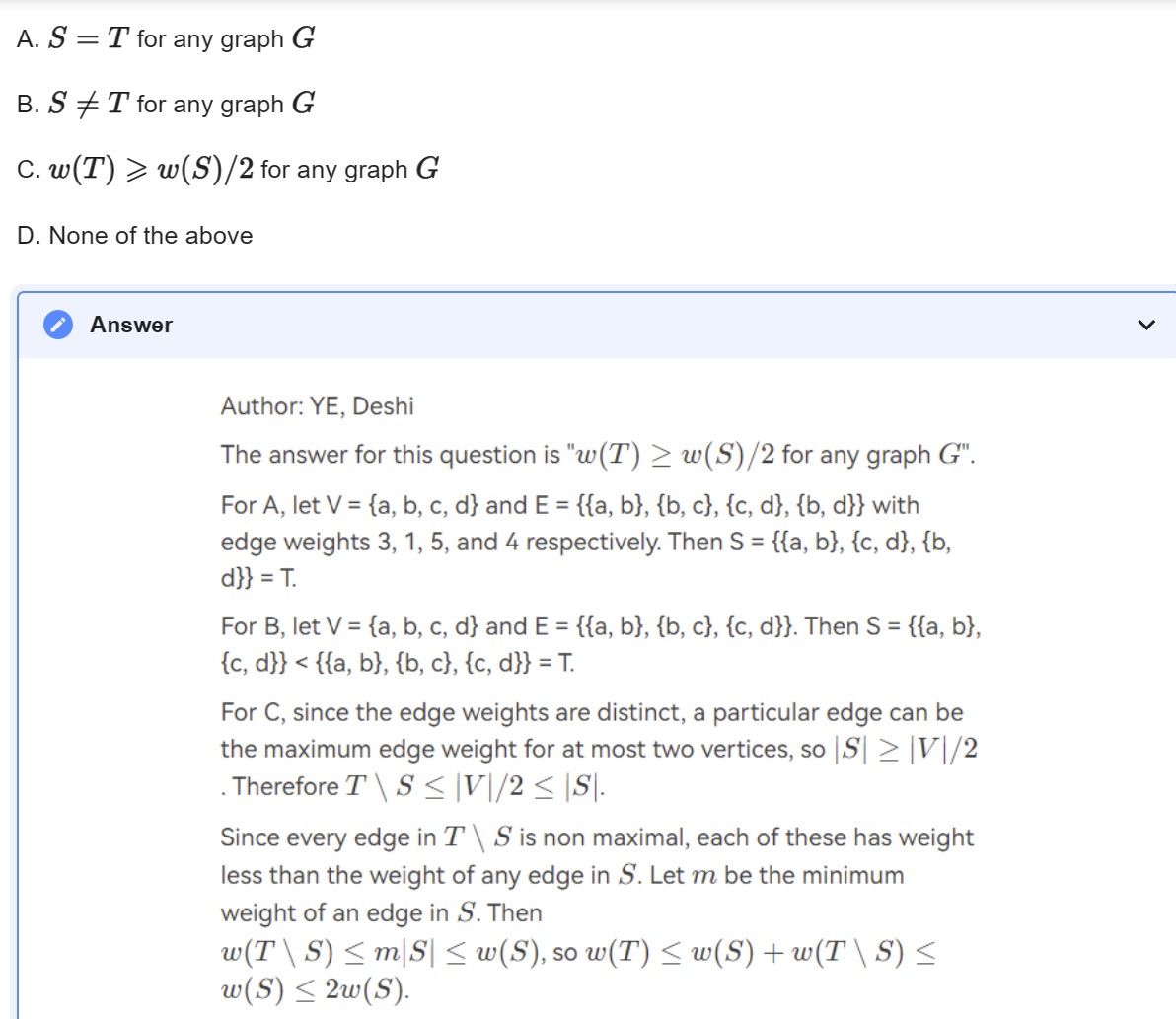
30、For the bin-packing problem: let S=∑i Si. Which of the following statements is FALSE?

A. The number of bins used by the next-fit heuristic is never more than⌈2S⌉

B. The number of bins used by the first-fit heuristic is never more than ⌈2S⌉

C. The next-fit heuristic leaves at most one bin less than half full

D. The first-fit heuristic leaves at most one bin less than half full

1. To approximate a maximum spanning tree T of an undirected graph G=(V,E) with distinct edge weights w(u,v) on each edge (u,v)∈E, let's denote the set of maximum-weight edges incident on each vertex by S. Also let w(E’)=∑(u,v)∈E’ w(u,v) for any edge set E’. Which of the following statements is TRUE?

期中：

1、If a problem can be solved by dynamic programming, it must be solved in polynomial time.  
F

[0-1背包问题](https://so.csdn.net/so/search?q=0-1%E8%83%8C%E5%8C%85%E9%97%AE%E9%A2%98&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/HGGshiwo/article/details/_blank)可以用DP解，但是复杂度不是多项式的, 原因是输入的数据不是多项式的。

2、All the languages can be decided by a non-deterministic machine.  
F  
不确定[图灵机](https://so.csdn.net/so/search?q=%E5%9B%BE%E7%81%B5%E6%9C%BA&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/HGGshiwo/article/details/_blank)可以用来验证NP问题的解是否是正确的，确定图灵机可以用来求解P问题。

NP hard问题无法通过不确定图灵机验证

1. Word stemming is to eliminate the commonly used words from the original documents.

F

参考PPT ADS03P8

1. In a red-black tree, the number of internal nodes in the subtree rooted at x is no more than 2bh(x)−1 where bh(x) is the black-height of x.

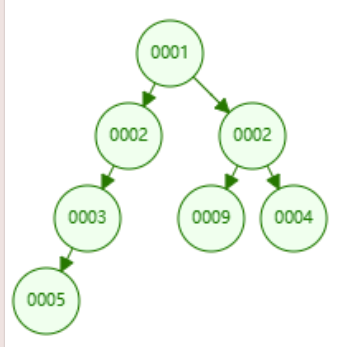
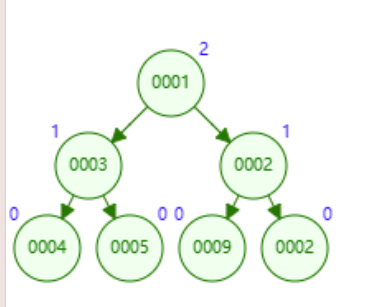
internal nodes指的是除了null的节点，这里应该是at least而不是no more

1. The right path of a skew heap can be arbitrarily long. 任意长

T

6、Given that problem A is NP-complete. If problem B is in NP and can be polynomially reduced to problem A, then problem B is NP-complete.

F

7、With the same operations, the resulting skew heap is always more balanced than the leftist heap

参考这两个分别是斜堆和左式堆

1. 左倾堆的合并操作和删除最小元素（DeleteMin）操作的最坏时间复杂度同样为 O(log n)。

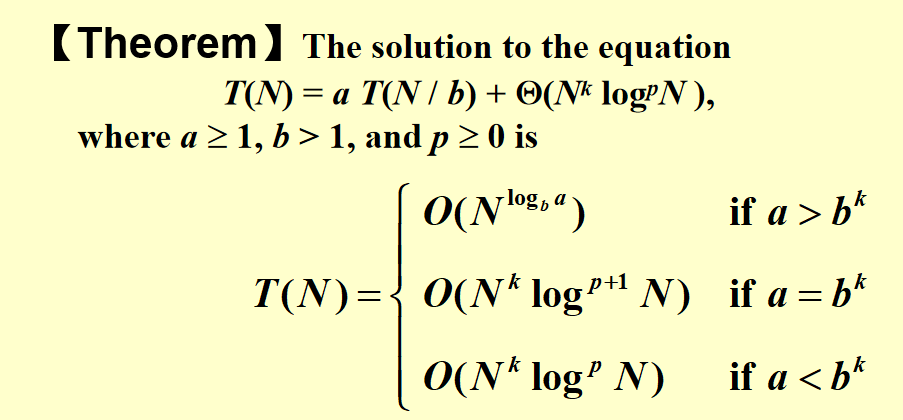
9、When solving a problem with input size N by divide and conquer, if at each stage the problem is divided into 4 sub-problems of equal size N/5, and the conquer step takes O(logN) to form the solution from the sub-solutions, then the overall time complexity is:

A.O(logN)

B.O(log2N)

C.O(N)

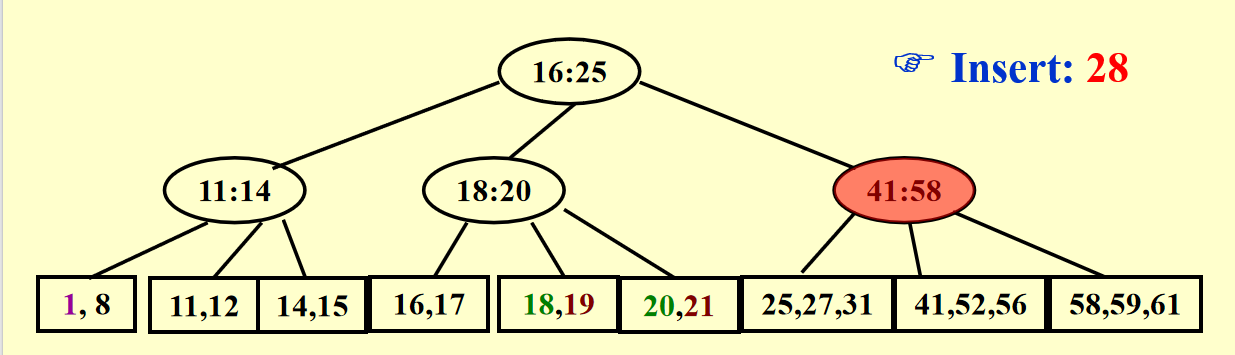
D.O(Nlog4/log5)

D

1. While accessing a term, hashing is faster than search trees.

T  
[哈希表](https://so.csdn.net/so/search?q=%E5%93%88%E5%B8%8C%E8%A1%A8&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/weixin_42887343/article/details/_blank)的插入、删除、查找的时间复杂度都是 O(1)；  
而平衡[二叉查找树](https://so.csdn.net/so/search?q=%E4%BA%8C%E5%8F%89%E6%9F%A5%E6%89%BE%E6%A0%91&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/weixin_42887343/article/details/_blank)的插入、删除、查找的时间复杂度都是 O(logn)

1. A B+ tree of order 3 with 21 numbers has at least \_0\_ nodes of degree 2.

 at most \_7\_ nodes of degree 2.

这种情况就是一个满树的形状，也就是有8个叶子节点，5\*3+3\*2=21个元素

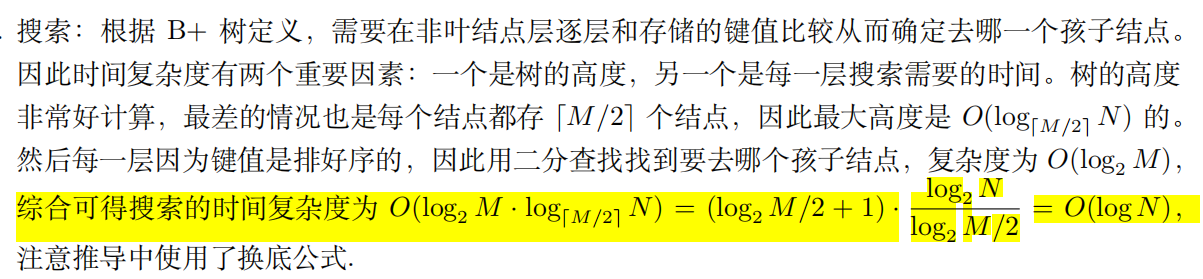
at most \_4\_\_ nodes of degree 3

这种情况就是一个1-3-9的形状,9个叶子结点3\*3+6\*2=21个元素

1. In backtracking, if different solution spaces have different sizes, start testing from the partial solution with the smallest space size would have a better chance to reduce the time cost.

T

1. The time bound of the FIND operation in a B+ tree containing N numbers is O(logN), no matter what the degree of the tree is.



1. 3-way-mergesort : Suppose instead of dividing in two halves at each step of the mergesort, we divide into three one thirds, sort each part, and finally combine all of them using a three-way-merge. What is the overall time complexity of this algorithm ?

T(N)=3T(N/3)+O(N)，故 O(NlogN)

14、Delete a node v from an AVL tree T1, we can obtain another AVL tree T2. Then insert v into T2, we can obtain another AVL tree T3. Which one(s) of the following statements about T1 and T3​is(are) true?

I、If v is a leaf node in T1, then T1  and T3 might be different.

II、If v is not a leaf node in T1, then T1 and T3 must be different.

III、If v is not a leaf node in T1, then T1 and T3 must be the same.

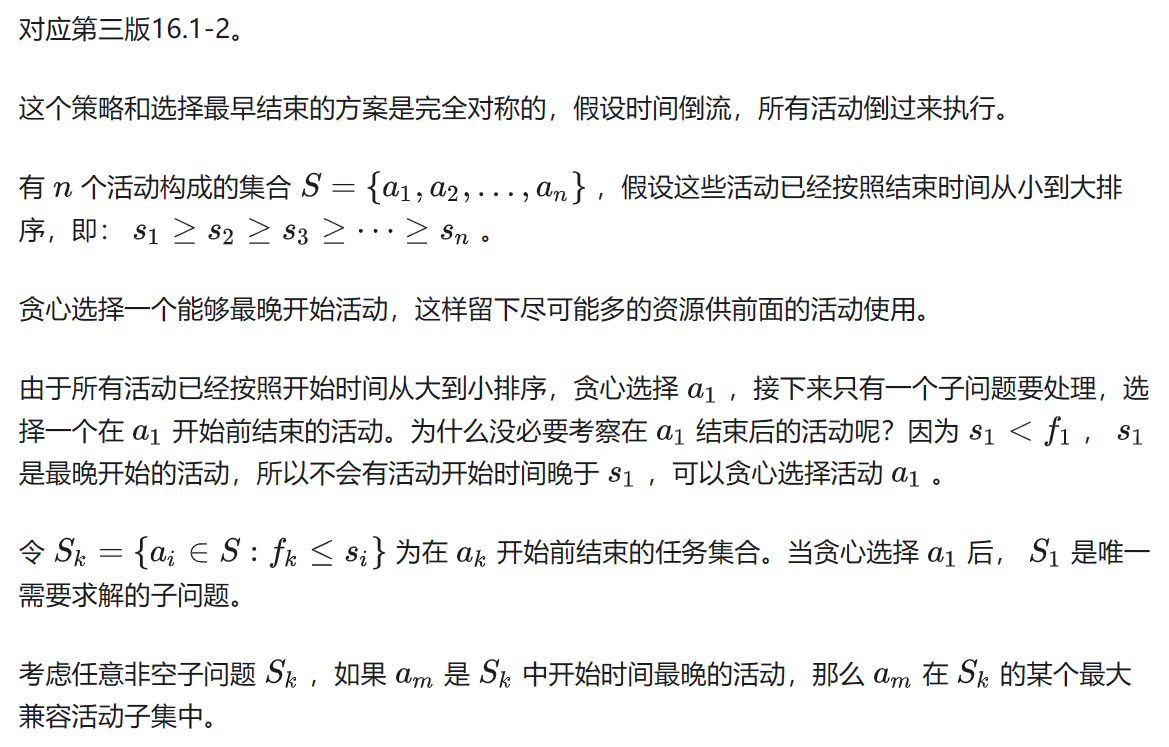
19-20函数题R5-2

15、Consdier the activity selection (interval scheduling) problem. In class, we have proved that an

optimal solution can be obtained by selecting the activities in increasing order of thier finishing

time. We can also obtain an optimal solution by selecting the activities in decreasing order of their start time.

T



1. Consider the recurrence T(n)=a\*\*T(n/b)+n\*\*d with T(1)=1. If a<b\*\*d, then T(n) is dominated (in asympototic sense) by the amount work at the root of the recursion tree. If a>b\*\*d, then T(n) is dominated (in asympototic sense) by the total amount work at the leaves of the recursion tree.

T

17、Consider an ordinary binary min-heap that supports INSERTION and DELETEMIN only. For each key x in the heap, denote its depth by d(x). We assume that the depth of the root is 1. We define a potential function Φ(D)=c\*∑x∈D\*\*d(x), where c is a constant that is sufficiently large. That is, the potential of a heap is proportional to the sum of the depths of the keys stored in it. Using this potential function Φ, what are the amortized costs of INSERTION and DELETEMIN? Your bounds should be as tight as possible.

A. O(logn) for INSERTION and O(logn) for DELETEMIN

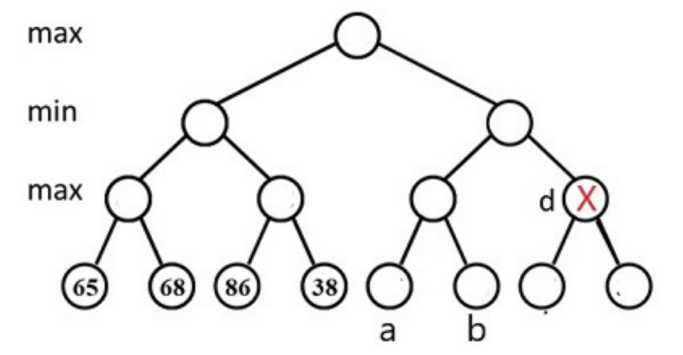
B. O(1) for INSERTION and O(logn) for DELETEMIN

C. O(logn) for INSERTION and O(1) for DELETEMIN

D. O(n) for INSERTION and O(n) for DELETEMIN

C

18、Consider the following game tree. If node d is pruned by α-β pruning algorithm, which of the following statements about the value of node a or node b must be correct?



A. both are greater than or equal to 68

B. both are less than or equal to 68

C. exactly one of them is greater than 68

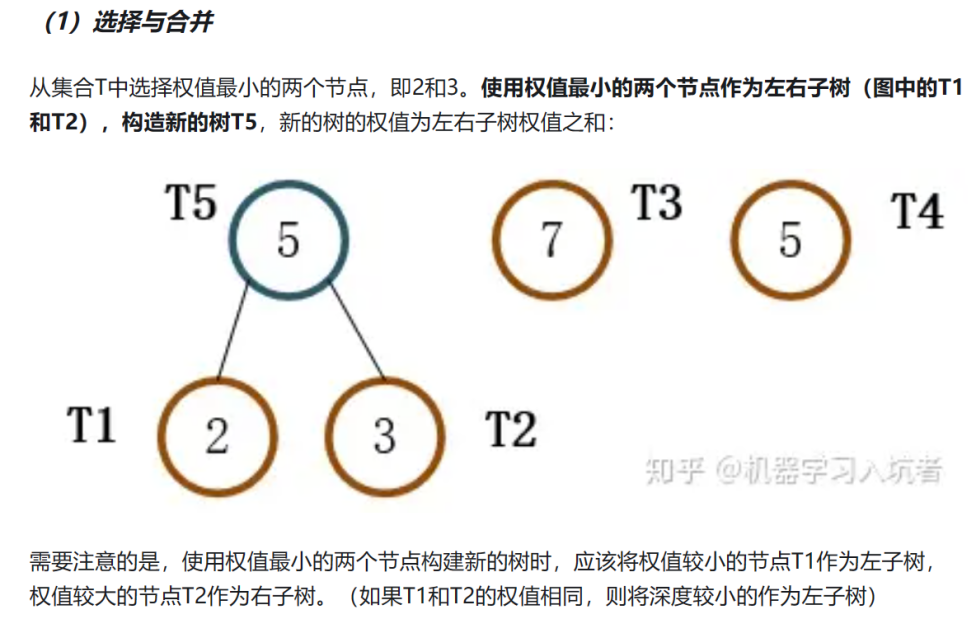
D. exactly one of them is less than 68

B

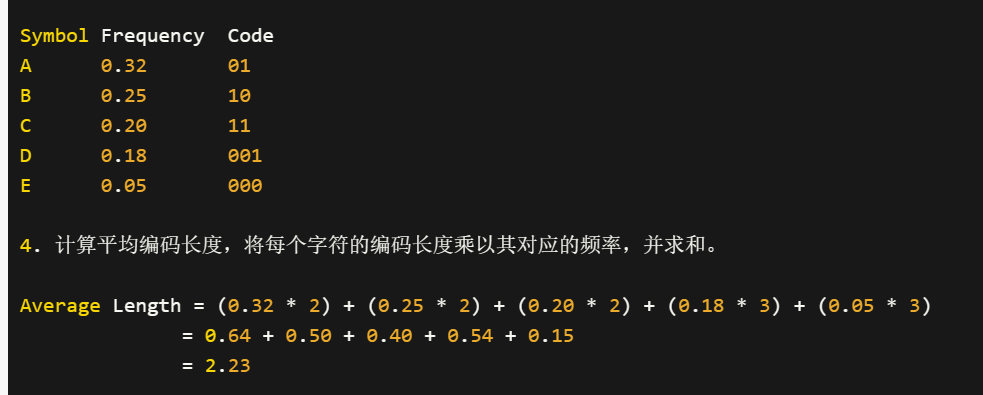
1. Consider the following symbol frequencies for a five-symbol alphabet:

A 0.32  
B 0.25  
C 0.20  
D 0.18  
E 0.05

What is the average encoding length of an optimal prefix code?



先合并DE-->0.23小于B所以合并这个节点和C，合并后比AB都大所以下一个合并AB



19、Consider the following pseudo-code.

strange(a1,…,a\*\*n):

1.if n≤2022 then return

2. strange(a⌊n/4+1⌋,…,a⌊3n/8⌋)

3.for i=1 to n:

4. print(a\*\*i)

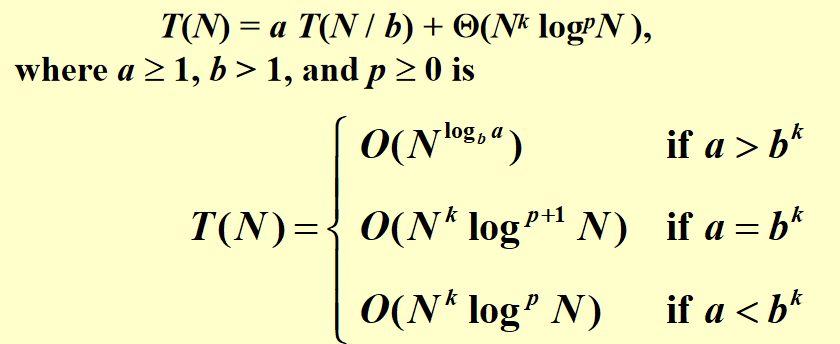
5.strange(a⌊3n/4+1⌋,…,a\*\*n)

What is the running time of this pseudo-code? Your answer should be as tight as possible. (Hint:

give a recurrence for this pseudocode and solve the recurrence with the method of your choice.

You may assume n is a power of 2.)

第2行的递归调用：strange(a⌊n/4+1⌋, ..., a⌊3n/8⌋)。这个递归调用的范围是从 a⌊n/4+1⌋ 到 a⌊3n/8⌋，大小为 (3n/8 - n/4 + 1)。第5行的递归调用：strange(a⌊3n/4+1⌋, ..., a\*\*n)。这个递归调用的范围是从 a⌊3n/4+1⌋ 到 a\*\*n，大小为 (n - 3n/4 + 1)。在第4行的循环中，我们迭代从 1 到 n 的整数，并执行 print(a\*\*i)，即打印 a 的 i 次方。因此，我们可以得到递归关系：T(n) = T((3n/8 - n/4 + 1)) + T((n - 3n/4 + 1)) + n

在递归关系 T(n) = T(n/8) + T(n/4) + n 中，我们有两个递归调用，并且问题的规模分别减小为原来的 1/8 和 1/4。因此，正确的参数应为：

a = 2，表示递归关系有两个递归调用。

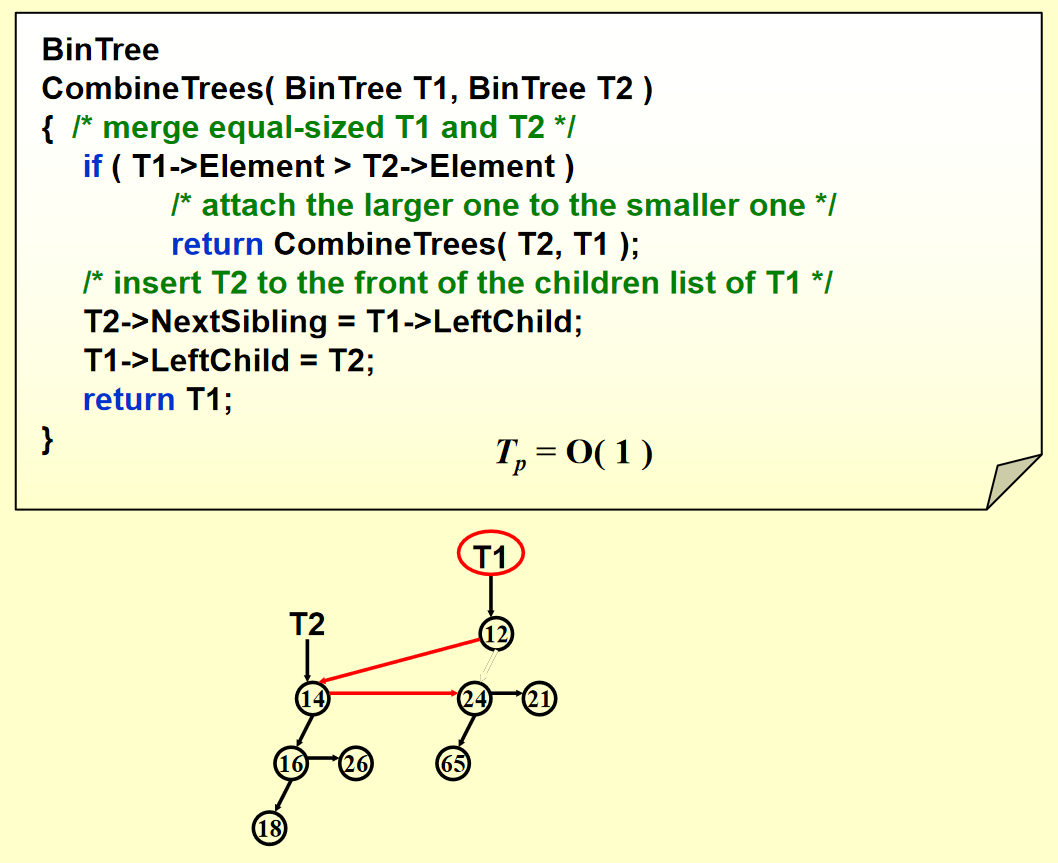
b = 8，表示问题的规模减小为原来的 1/8。

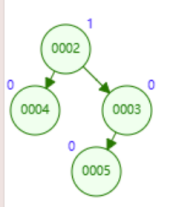
f(n) = n，表示其他操作的运行时间。

答案是O(N)

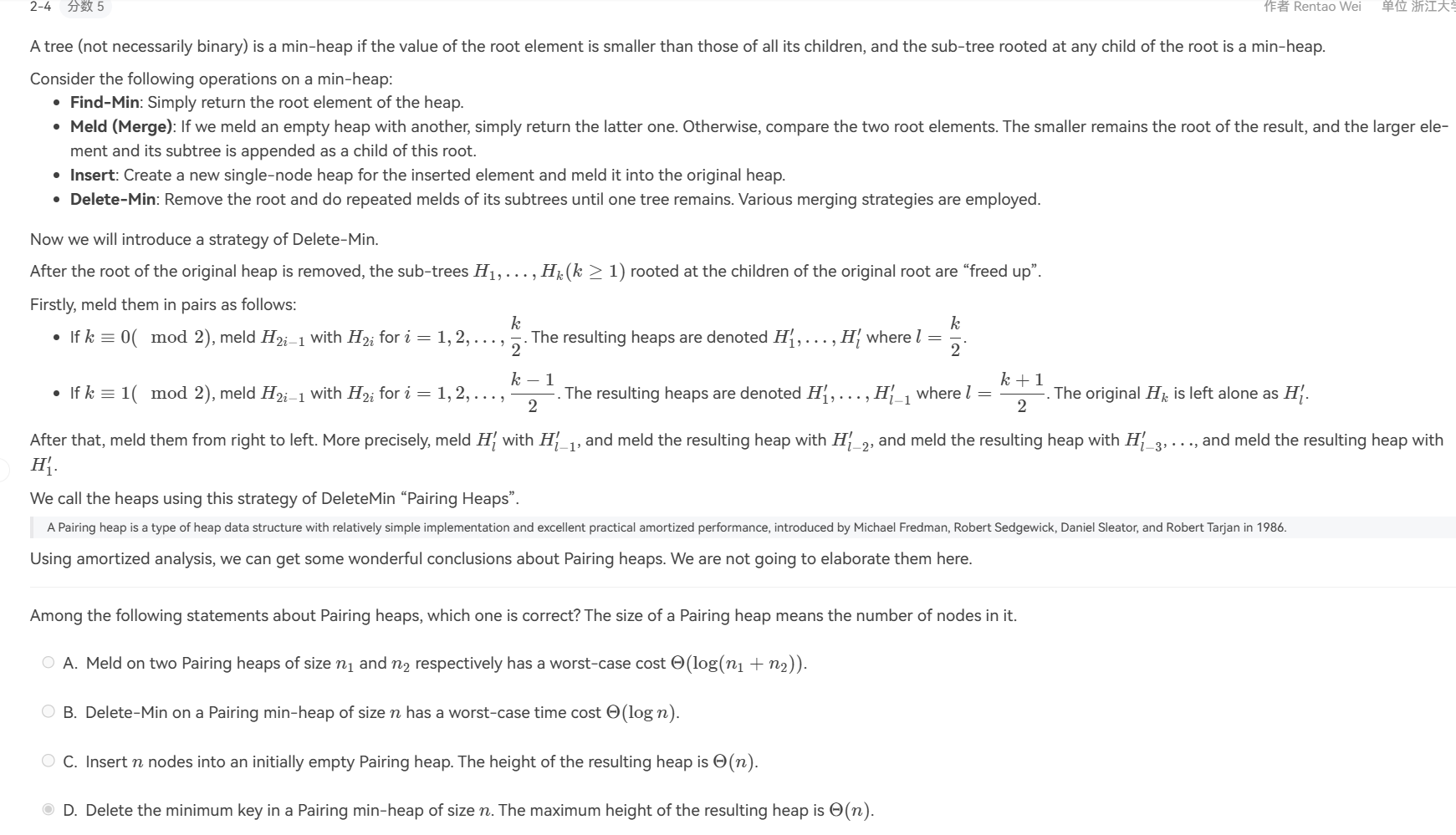
20、To implement a binomial queue, the subtrees of a binomial tree are linked in increasing sizes

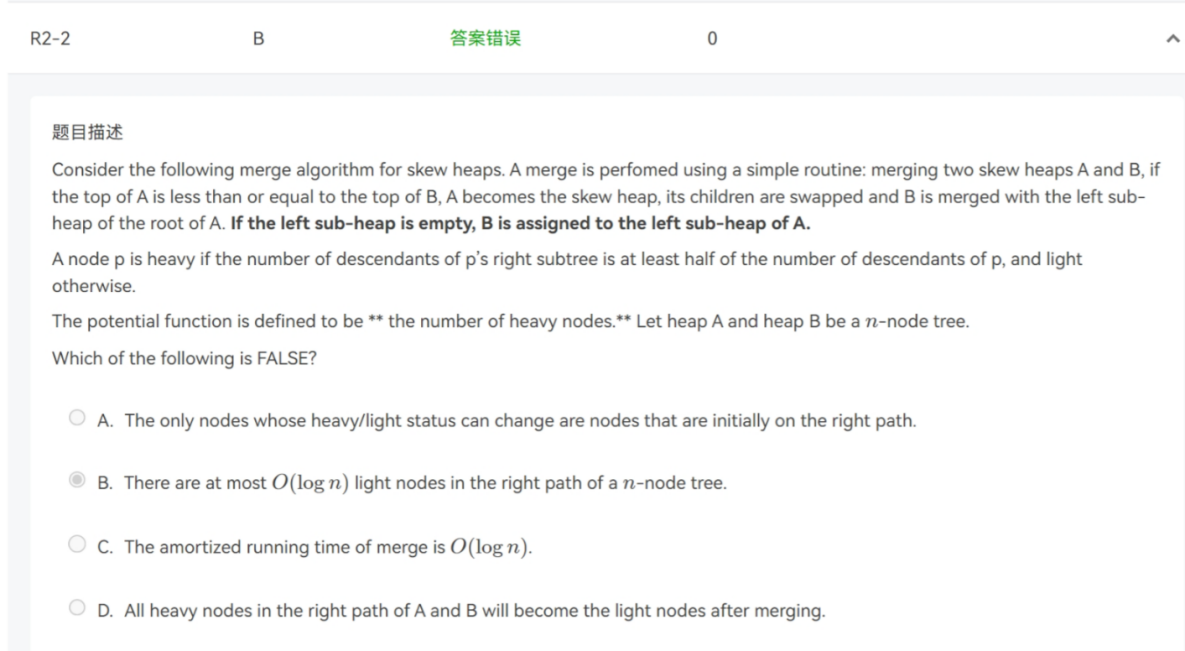
F降序排直接merge就好了，升序排还得遍历找到size相同的那个



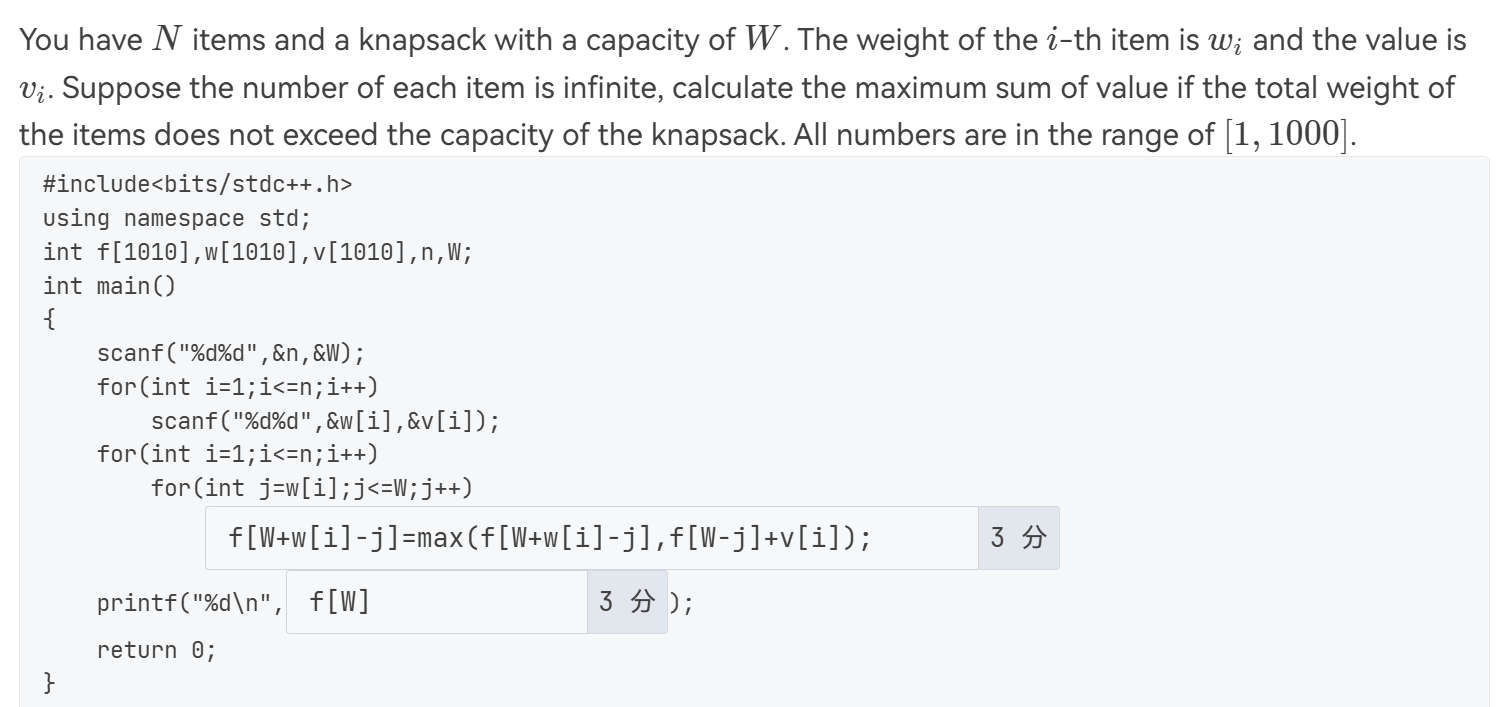
1. yy函数题5-2
2. 动态规划求最大子串和
3. After inserting a node into a Leftist heap H (which is equivalent to merging a one-node Leftist heap with H), we need to swap the children of at most 1 node to make the resulting tree a Leftist heap.

T

24、



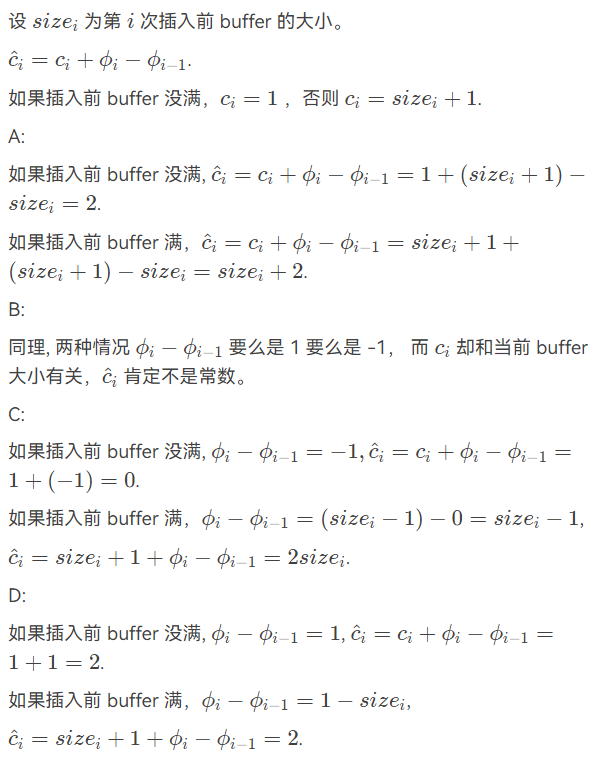
D 万一A的左子树就是空的话，是不是直接停止合并了



一维背包解法：第1空：f[j]=max(f[j],f[j-w[i]]+v[i]); 第2空：f[W]

朋辈辅学：

01-2-4

01-3-4

02-1-7

02-3-4

03-1-5

15-16期末

1. An AVL tree with the balance factors of all the non-leaf nodes being 0 must be a perfect binary tree

√

1. The worst-case running time is equal to the expected running time within constant factors for any randomized algorithm

×

1. The worst-case running time is equal to the expected running time within constant factors for any randomized algorithm

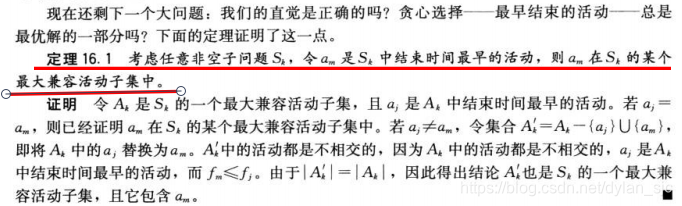
× 对于随机算法，其运行时间的期望值可以被视为算法在所有可能输入上的平均性能。然而，最坏情况下的运行时间可能会远远超过期望运行时间

1. With the same operations, the resulting leftist heap is always more balanced than the skew heap

× 这两没有谁比谁更加balanced都存在反例

1. Let S be the set of activities in Activity Selection Problem. Then there must be some maximum-size subset of mutually compatible activities of S that includes the earliest finish activity am

√

1.  To merge 55 runs using 3 tapes for a 2-way merge, the original distribution (30, 25) is better than (34, 21).

√ 斐波那契数的外部排序最优，如果用original distribution (30, 25)会导致最后出现一个tape上有5个顺串，这需要拷贝一次拆分为2和3个串继续进行排序导致额外的时间

1. Recall that the worst-case time complexities of insertions and deletions in a heap of size N are both O(logN). Then, without changing the data structure, the amortized time complexity of insertions in a heap is also O(logN), and that of deletions is O(1).

√ 参考上面的题目，将势能函数定义为每个节点的深度和，Ci^ = Ci加上势能差，这样insert的时候势能变化为logn且实际开销Ci为logn，∑Ci^=nlogn所以摊还为logn，但是delete的实际开销O(logn)，势能变化是1-logn所以摊还是O(1)

1. Suppose ALG is an α-approximation algorithm for an optimization problem Π whose approximation ratio is tight. Then for every ϵ>0 there is no (α−ϵ)-approximation algorithm for Π unless P = NP.

× 对于一种算法而言，近似比为 α ，那么 ∀β>α ，都可以说 β 是其近似比。如果 α 是 tight 的，则 α 是一个下确界。但这都只是对这一种算法的分析，一个 tight 的近似比只能说明你对这种算法的分析到位了，而不能说明这个问题没有更好的算法。这里完全是两码事

1. The decision problem HALTING returns TRUE, if, for a given input I and a given (deterministic) algorithm A, A terminates, otherwise it loops forever. The HALTING problem is NP-complete.

× 首先回顾归约：问题A可约化为问题B”的一个隐含的含义是B的时间复杂度 >= A的时间复杂度，这里的意义是既然更复杂的B问题能被解决，那么相对简单的A问题自然也能被解决了。

首先假设有这样一个NP问题，所有的NP问题都可以归约为它，那么只要该问题能找到多项式级的解法，则所有NP问题也都能通过规约为它从而找到多项式级解法。由此可见，NPC问题是最复杂的NP问题，但NP问题不一定是NPC问题。事实上，NPC问题并不是只有一个，而是有很多个。

停机问题是NP-Hard但停机问题是不可判的，那它当然不是NP问题，所以不是NP-Complete 但对于SAT这样的NP-Complete问题，却可以多项式归约到停机问题

1. When measuring the relevancy of the answer set of a search engine, the precision is low means that most of the relevant documents are not retrieved.

× P = RR / (RR + IR) the precision is low 意味着返回的IR很多即很多不相关的被返回

1. For a binomial queue, delete-min and merging take logn time on average

√

1. Which one of the following statements about the Maximum Finding problem is true?
2. Parallel random sampling algorithm can run in O(1) time and O(N) work with very high probability.
3. There exists a serial algorithm with time complexity being O(logN).

C. No parallel algorithm can solve the problem in O(1) time.

D. When partitioning the problem into sub-problems and solving them in parallel, compared with 根号N, choosing loglogN as the size of each sub-problem can reduce the work load and the worst-case time complexity. 看PPT第17页时间是一样的但是W(n)不一样

A

1. Comparing to leftist heaps, skew heaps are always more efficient in space

√ 因为不需要存储npl数据

1. Max-cut problem: Given an undirected graph G = (V, E) with positive integer edge weights we, find a node partition (A, B) such that w(A, B), the total weight of edges crossing the cut, is maximized. Let us define S' be the neighbor of S such that S' can be obtained from S by moving one node from A to B, or one from B to A. We only choose a node which, when flipped, increases the cut value by at least w(A,B)/|V|. Then which of the following is true?   
   A. Upon the termination of the algorithm, the algorithm returns a cut (A, B) so that 2.5 w(A, B) >= w(A\*, B\*), where (A\*, B\*) is an optimal partition.  
   B. The algorithm terminates after at most O(log|V| log W) flips, where W is the total weight of edges.  
   C. Upon the termination of the algorithm, the algorithm returns a cut (A, B) so that 2 w(A, B) >= w(A\*, B\*)  
   D. The algorithm terminates after at most O(|V|^2) flips.

A

1. Given a 3-SAT formula with k clauses, in which each clause has three variables, the MAX-3SAT problem is to find a truth assignment that satisfies as many clauses as possible. A simple randomized algorithm is to flip a coin, and to set each variable true with probability 1/2, independently for each variable. Which of the following statements is FALSE?
2. The probability that a random assignment satisfies at least 7k/8 clauses is at most 1/(8k).
3. If we repeatedly generate random truth assignments until one of them satisfies ≥7k/8 clauses, then this algorithm is a 8/7-approximation algorithm.
4. The expected number of clauses satisfied by this random assignment is 7k/8.
5. For every instance of 3-SAT, there is a truth assignment that satisfies at least a 7/8 fraction of all clauses.

A 是at least

1. Suppose Q is a problem in NP, but not necessarily NP-complete.

A polynomial-time algorithm for Q would sufficiently imply a polynomial-time algorithm for SAT.

×

A polynomial-time algorithm for SAT would sufficiently imply a polynomial-time algorithm for Q.

√

如果一个npc问题能在多项式时间内解决那么每一个np问题也可以在多项式时间内解决

1. Which of the following statement is true ?

A. Let A and B be optimization problems where it is known that A reduces to B in polynomial time. Additionally, suppose that there exists a polynomial-time 2-approximation for B. Then there must exist a polynomial time 2-approximation for A.

B. There exists a polynomial-time 2-approximation algorithm for the general Traveling Salesman Problem.

C. A randomized algorithm for a decision problem with one-sided-error and correctness probability 1/3 (that is, if the answer is YES, it will always output YES, while if the answer is NO, it will output NO with probability 1/3) can always be amplified（放大） to a correctness probability of 99%.

D. Suppose that you have two deterministic online algorithms, A1 and A2, with competitive ratios (the approximation ratio for an online algorithm is called competitive ratio) c1 and c2 respectively. Consider the randomized algorithm A∗ that flips a fair coin once at the beginning; if the coin comes up heads, it runs A1 from then on; if the coin comes up tails, it runs A2 from then on. Then the expected competitive ratio of A∗ is at least min{c1, c2}.

C

16-17

1. Let S be the set of activities in Activity Selection Problem. The greedy rule of "collecting the activity that starts the latest" is correct for finding a maximum-size subset of mutually compatible activities of S.

√ 从后往前看，最晚开始就是从前往后的最早结束问题

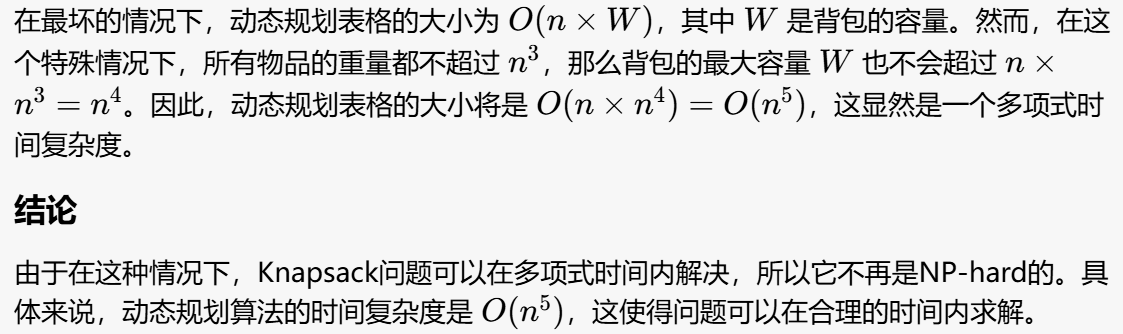
1. Recall the amortized analysis for Splay Tree and Leftist Heap, from which we can conclude that the amortized cost (time) is never less than the average cost (time)

×

1. A skew heap is a heap data structure implemented as a binary tree. Skew heaps are advantageous because of their ability to merge more quickly than balanced binary heaps. The worst case time complexities for Merge, Insert, and DeleteMin are all O(N), while the amorited time complexities for Merge, Insert, and DeleteMin are all O(logN)

√

1. Consider a Knapsack problem with n items. If no items have a size larger than n3, then it is no longer NP-hard√



1. Assume that Problem X is reduced to Problem Y in polynomial time, where Y is NP-hard. Moreover, Y adimts a ρ-approximation algorithm, and there is no (ρ−ϵ)-approroximation algorithm unless P=NP. Which one of the following statements is TRUE?

A. Neither of the other three is correct.

B. X is NP-hard too.

C. X has no (ρ−ϵ)-approximation algorithm unless P=NP.

D. X has a ρ-approximation algorithm.

A

1. If there are 28 nodes in an AVL tree, then the maximum depth of the tree is \_\_. The depth of an empty tree is defined to be 0.

6计算斐波那契并且可以直接画图

1. B+树的插入再看一遍，尤其是分裂；还有红黑树的插入和删除
2. Let us consider the problem of finding a large cut in an undirected graph G=(V,E) which has n vertices and m edges. A cut is a partition of the vertices into two disjoint sets, and the value of a cut is the weight of all edges crossing from one side of the partition to the other. Here we consider the case where all edges in the graph have the same weight 1. Suppose that a partition of V into two disjoint sets A and B is given by a randomized algorithm, in which each vertex is randomly and independently assigned to one of the two sets.Which of the following statements is FALSE?
3. The expected approximation ratio of the algorithm is at most 2.
4. The probability of finding a cut with value at least m/2 is less than 2/(m+2).

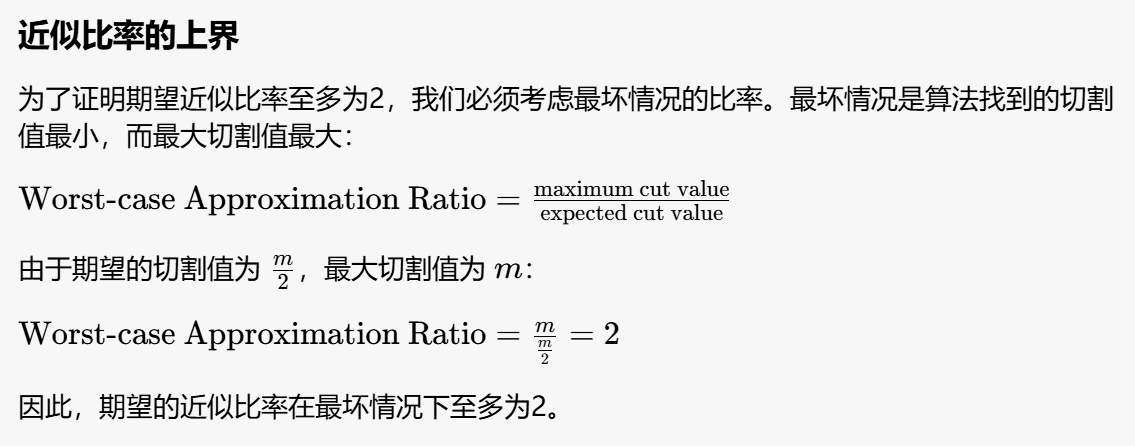
C. The expected number of edges in the partition generated by the algorithm is m/2.

D. There exists a partition A and B with at least m/2 edges connecting the set A to the set B.

B

每条边有 1/2 的概率被切割，期望切割值为 m/2。根据Chernoff界限，找到一个比期望值 m/2 更小的概率是非常小的。实际找到一个至少为 m/2 的切割值的概率应远大于2/(m+2)

这里的A解释如下：

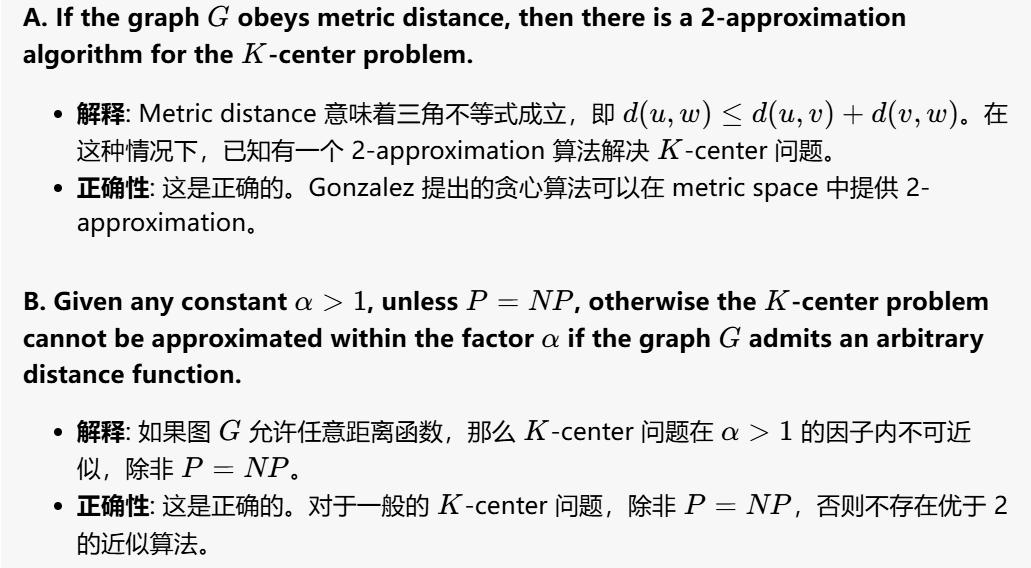
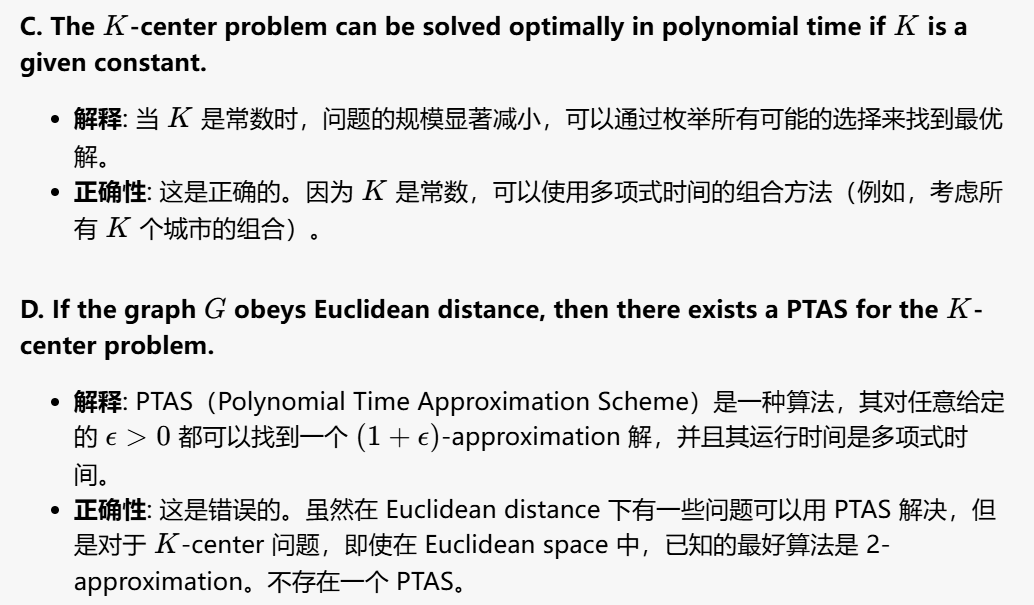


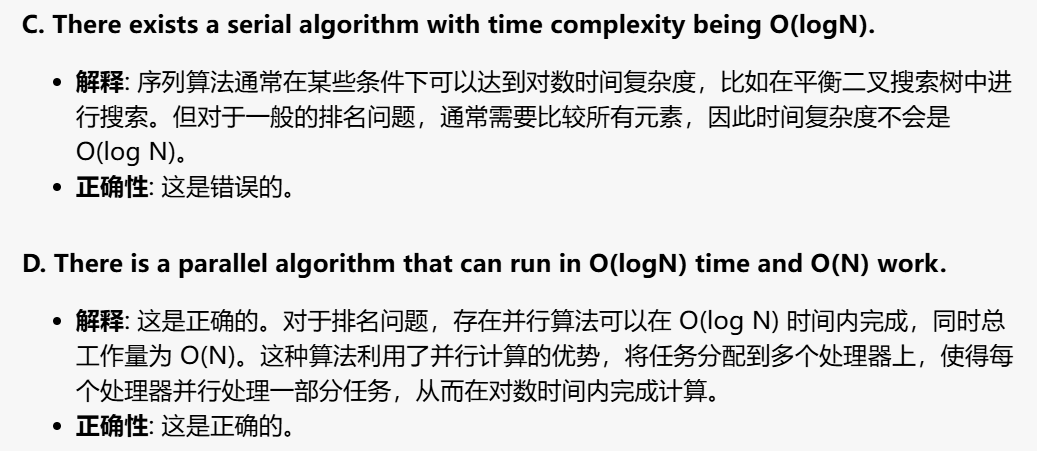
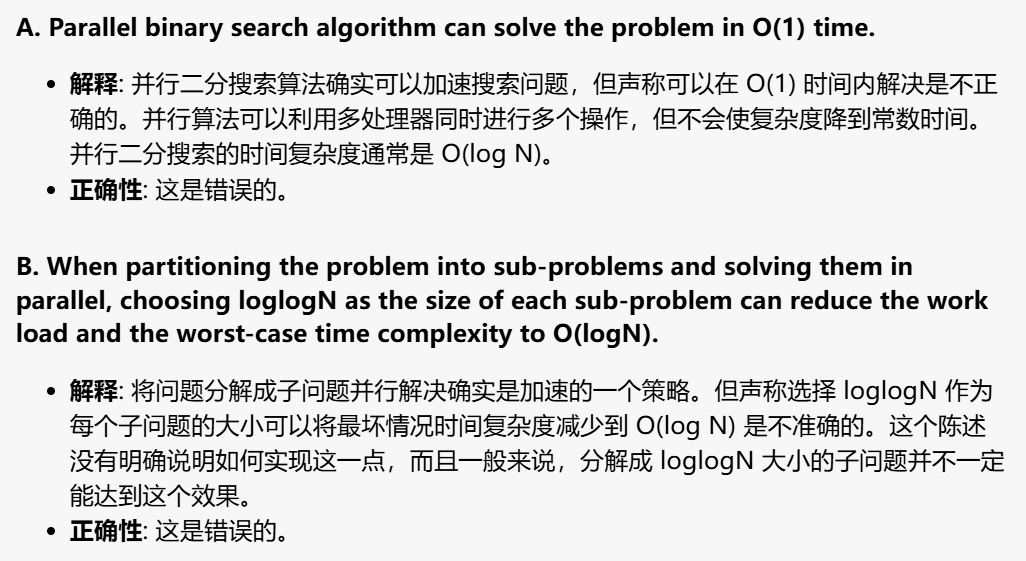
1. Spanning Tree Problem: Given an undirected graph G = (V, E), where |V | = n and |E| = m. Let F be the set of all spanning trees of G. Define d(u) to be the degree of a vertex u ∈ V. Define w(e) to be the weight of an edge e ∈ E.  
   We have the following three variants of spanning tree problems:  
   - (1) Max Leaf Spanning Tree: find a spanning tree T ∈ F with a maximum number of leaves.找到一个生成树 𝑇T 使得树中的叶子数最大  
   - (2) Minimum Spanning Tree: find a spanning tree T ∈ F with a minimum total weight of all the edges in T. 找到一个生成树 T 使得树中所有边的总权重最小  
   - (3) Minimum Degree Spanning Tree: find a spanning tree T ∈ F such that its maximum degree of all the vertices is the smallest. 找到一个生成树 T 使得所有顶点的最大度数最小

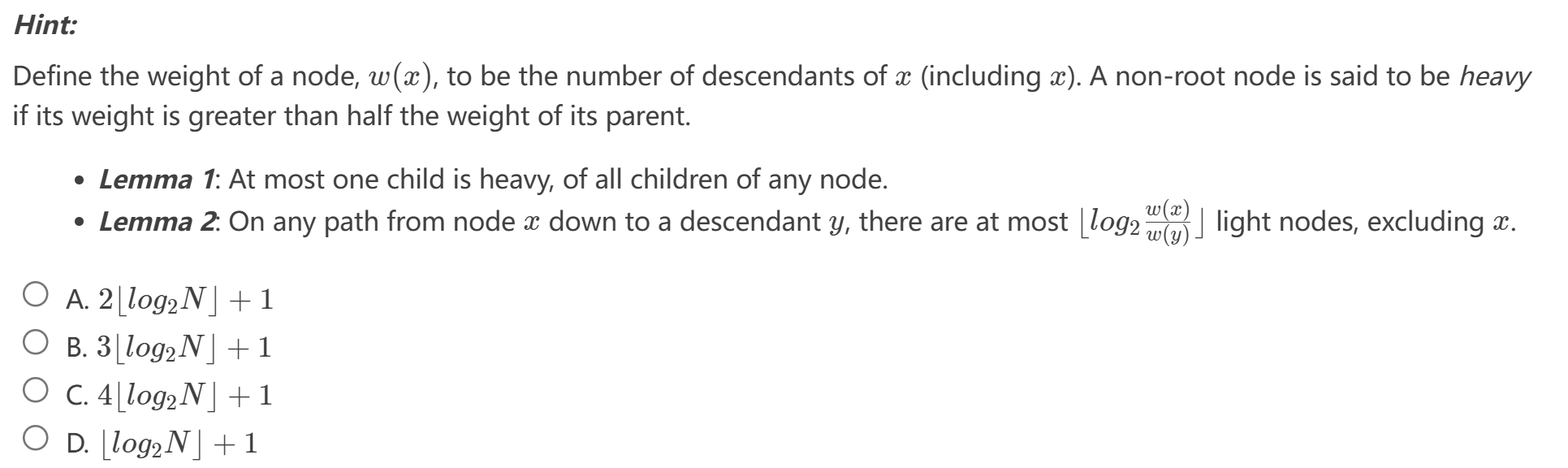
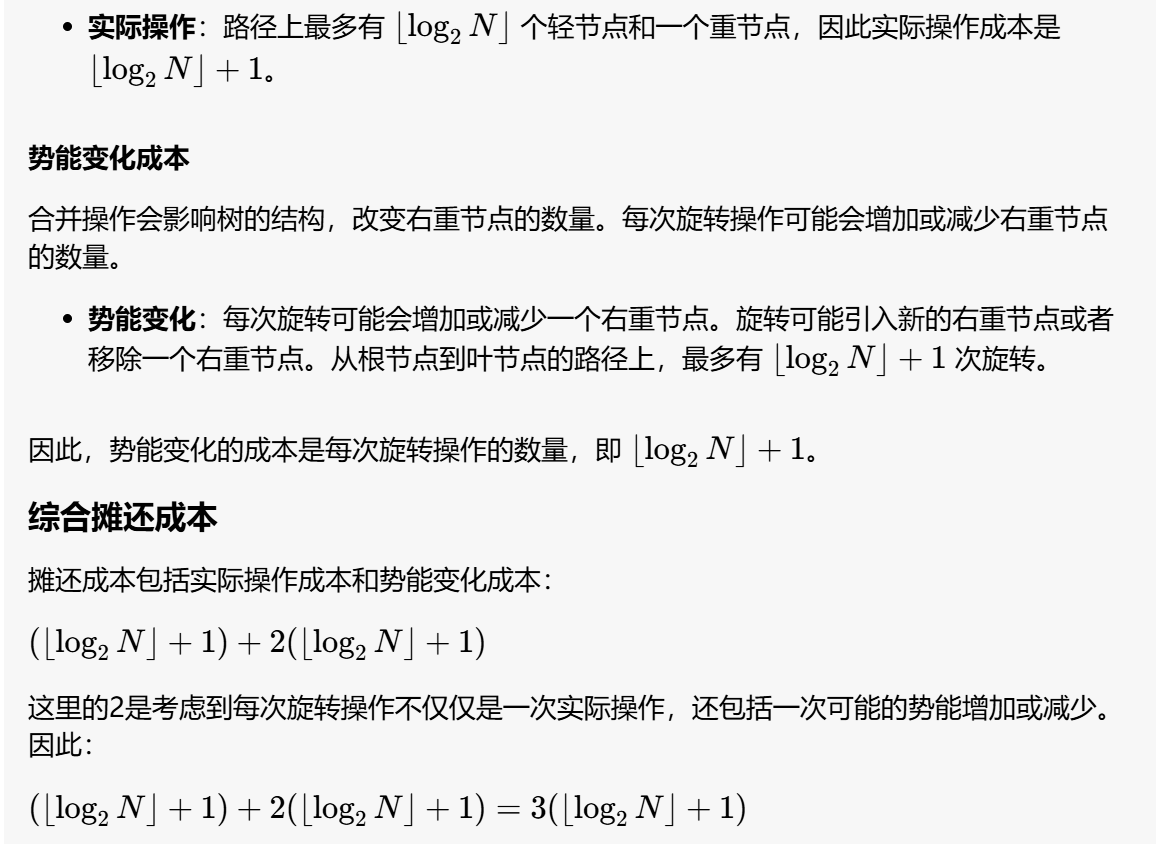
For a pair of edges (e, e') where e ∈ T and e' ∈ (G-T) such that e belongs to the unique cycle of T∪ e', we define edge-swap(e, e') to be (T-e)∪ e'.  
Here is a local search algorithm: 交换生成树中的边和不在生成树中的边  
 T = any spanning tree in F;  
 while (there is an edge-swap(e, e') which reduces Cost(T)) {  
 T = T - e + e';  
 }  
 return T;  
Here `Cost(T)` is the number of leaves in `T` in Max Leaf Spanning Tree; or is the total weight of `T` in Minimum Spanning Tree; or else is the minimum degree of `T` in Minimum Degree Spanning Tree.  
Which of the following statements is TRUE? [B]  
A. The local search always return an optimal solution for Max Leaf Spanning Tree  
B. The local search always return an optimal solution for Minimum Spanning Tree  
C. The local search always return an optimal solution for Minimum Degree Spanning Tree  
D. For neither of the problems that this local search always return an optimal solution



1. K-center problem: Given N cities with specified distances, one wants to build K warehouses in different cities and minimize the maximum distance of a city to a warehouse.Which of the following is false?
2. If the graph G obeys metric distance, then there is a 2-approximation algorithm for the K-center problem.
3. Given any constant α>1, unless P = NP, otherwise the K-center problem cannot be approximated within the factor α if the graph G admits an arbitrary distance function.
4. The K-center problem can be solved optimally in polynomial time if K is a given constant.
5. If the graph G obeys Euclidean distance, then there exists a PTAS for the K-center problem



1. Which one of the following statements about the Ranking problem is true? (Assume that both arrays contain N elements.)
2. Parallel binary search algorithm can solve the problem in O(1) time.
3. When partitioning the problem into sub-problems and solving them in parallel, choosing loglogN as the size of each sub-problem can reduce the work load and the worst-case time complexity to O(logN).
4. There exists a serial algorithm with time complexity being O(logN).
5. There is a parallel algorithm that can run in O(logN) time and O(N) work.
6. In proving the amortized bound of a Merge operation in skew heaps, the potential of a skew heap is defined to be the total number of right heavy nodes. Then we can prove that, in an N-node skew heap, the amortized cost for a Merge operation is exactly ？

B

19-20

1. If a leftist heap can be implemented recursively, so can its counterpart skew heap

T

1. Amortized bounds are weaker than the corresponding worst-case bounds, because there is no guarantee for any single operation. But amortized bounds are stronger than the equivalent average-case bounds, because average-case cannot guarantee any Moperations will behave as M\* average running time.
2. Let a=(a1,a2,…,ai,…,aj,…,an) denote the list of elements we want to sort. In the quicksort algorithm, if the pivot is selected uniformly at random. Then any two elements get compared at most once and the probability of ai and aj being compared is 2/(j−i+1) for j>i, given that ai or aj is selected as the pivot.

F, 假如有四个元素，第3个被选中。如果a1>a3, 如果a4<a3, 那么a1和a4发生交换，然后算法就停止了，不会比较a2和a3，此时而a1>a3概率为1/2，a4<a3的概率为1/2, 因此概率为1/4

具体要算的话，应该是N-j-1个元素中大于j的元素个数比i-1个中要多的概率。算起来好像比较麻烦。

yds后来解释了一下，如果a1,到aN是有序的，那么是正确的，因为只需要考虑i和j被分到不同的组，因此是i和j之间的这些元素中选择了i或者j的概率。

1. If L1 ≤p L2 and L2∈NP, then L1∈NP

T

1. To evaluate the Prefix-Sums of a sequence of 16 numbers by the parallel algorithm with Balanced Binary Trees, C(4,1) is found before C(2,2).

F C是从上往下计算的，第一个数字是层数。

To evaluate the sum of a sequence of 16 numbers by the parallel algorithm with Balanced Binary Trees, B(1,6) is found before B(2,1).

T B是从下往上算的

6.

21-22

1. Amortized analysis is a technique to provide an upper bound on the actual cost of a sequence of operations T
2. For the Turnpike reconstruction algorithm of N points, assuming that the distance set D is maintained as an AVL tree, the running time is O(N2logN) if no backtracking happens

T 因为N个点所以D内有N(N-1)/2个数，遍历AVL tree的高度logN，每经过一个点将D中的响应距离删掉（N2的数据的遍历）

1. After merging two Leftist Heaps H1 and H2, the NPL of the resulted Leftist Heap will be no more than min(NPL of H1, NPL of H2)+1

F 书本上的例子加一个节点就可以举出反例

1. Recall that, to solve the closest pair problem, the first step of the divide-and-conquer algorithm divides the point set into L and R according to the x-coordinate. Is the following statement true of false? *In the combine step of this algorithm, it is always able to find the closest pair with one point in L and the other in R.*

F

1. Consider the randomized quicksort. We have proved that it runs in O(nlogn) time in expectation even for the worst input. Is the following statement true of false? *There exists some good inputs on which the expected running time of randomized quicksort is O(n) where n is the input size*

F 最好情况log底为2，最差4/3

1. If some NP-complete problem can be solved in polynomial time, then P = NP

T 既然所有的NP问题都能约化成NPC问题，那么只要任意一个NPC问题找到了一个多项式的算法，那么所有的NP问题都能用这个算法解决了，NP也就等于P 了

1. Which one of the following statements about the Ranking problem is TRUE? (Assume that both arrays contain N elements)

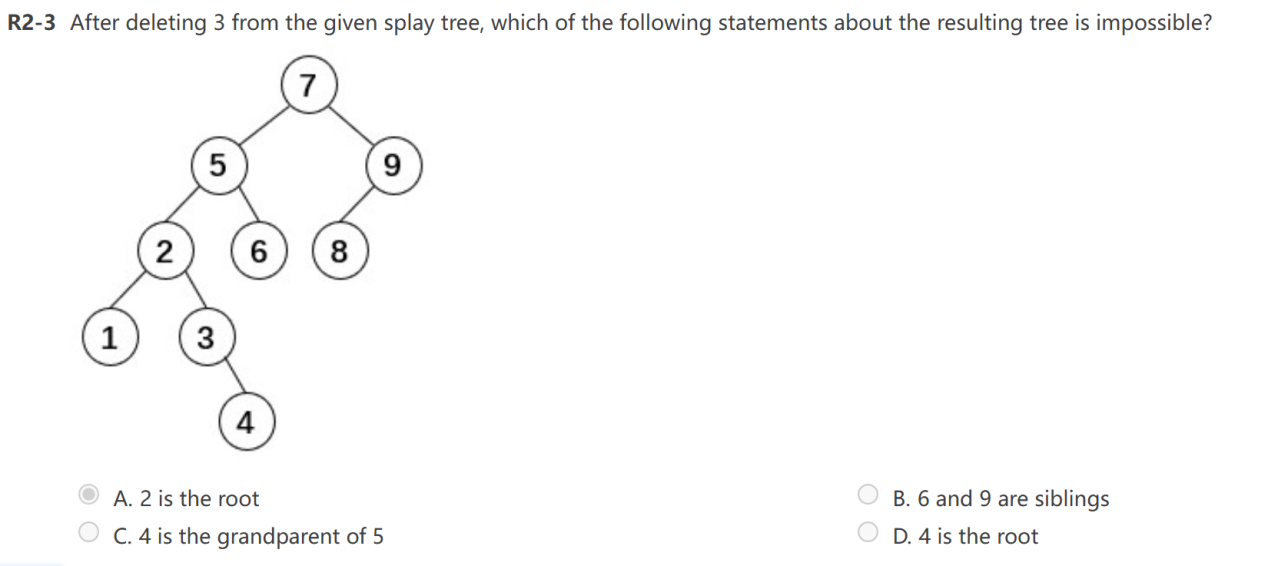
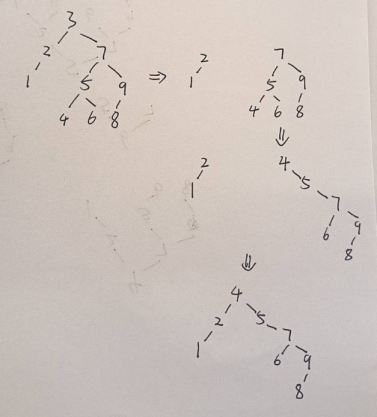
A.Serial ranking algorithm has better time complexity comparing with the binary search algorithm.

B.It can be used in merging problem and make the merging problem solved in O(1) time.

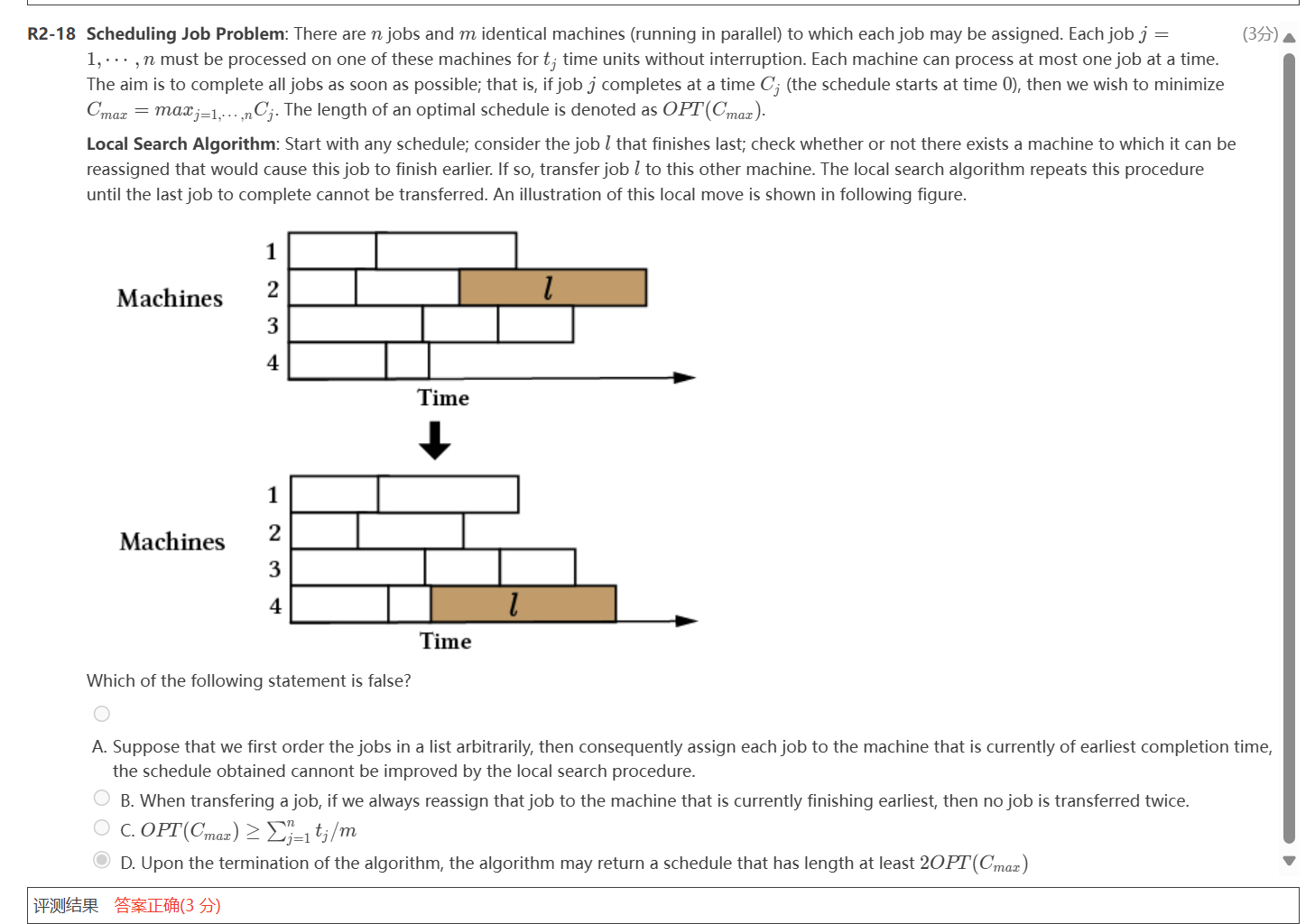
C.Parallel binary search algorithm will solve the problem in O(logN) time with O(logN) work load.

D.Using binary search algorithm to solve the problem will make the time complexity be O(NlogN)

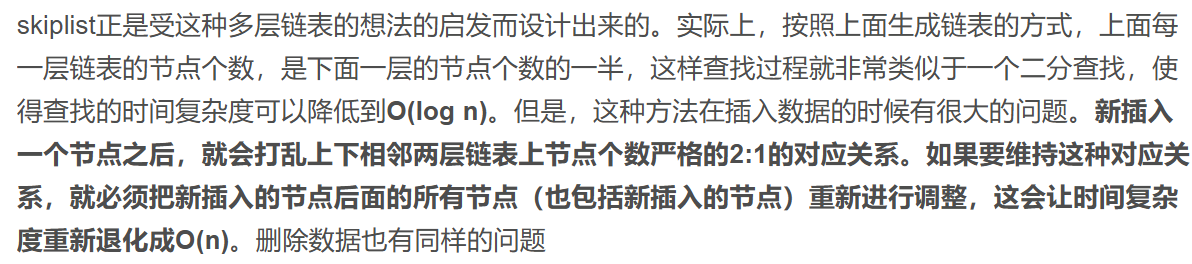
B 这里指的是已经有了rank后解决merge问题的时间是O(1)

1. 这题选C，注意在删除后如果要把比如4挪到根节点是以splay树的形式移动，即下图
2. 多处理机任务调度问题，近似比为2-1/m其中m为机器数量，证明见网址

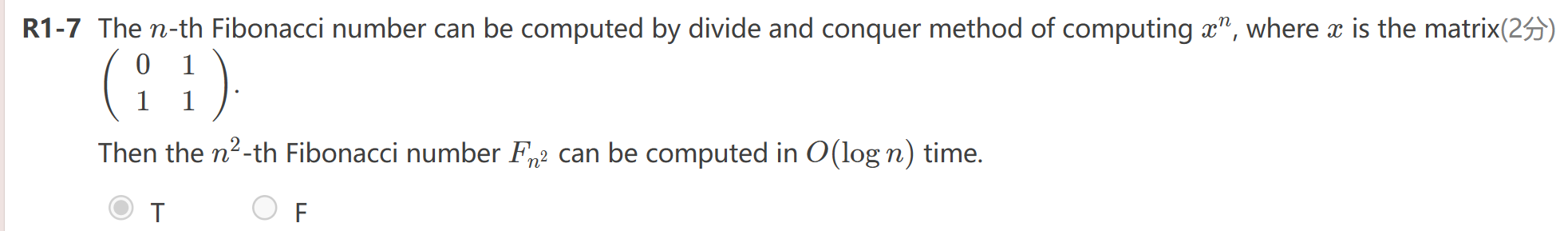
<https://www.cnblogs.com/cy0628/p/14016813.html>



20-21

1. If we build a skip list for n elements, then we can search or delete any element in O(logn) time in the worst case
2. For an optimization problem, given a neighborhood, if its local optimum is also a global optimum, one can reach an optimal solution with just one step of local improvements.

F 因为neighbor不一定是一步就能到的。定义里允许x步到达的邻居

1. 斐波那契数列矩阵求法：二阶矩阵二阶矩阵，8次乘法，4次加法就搞定了，所以说T(n)=T(n/2)+8O(1)
2. a binary tree T is full if each node is either a leaf or possesses exactly two child nodes. So A binary tree that is not full cannot correspond to an optimal prefix code.

T 不是满树！！

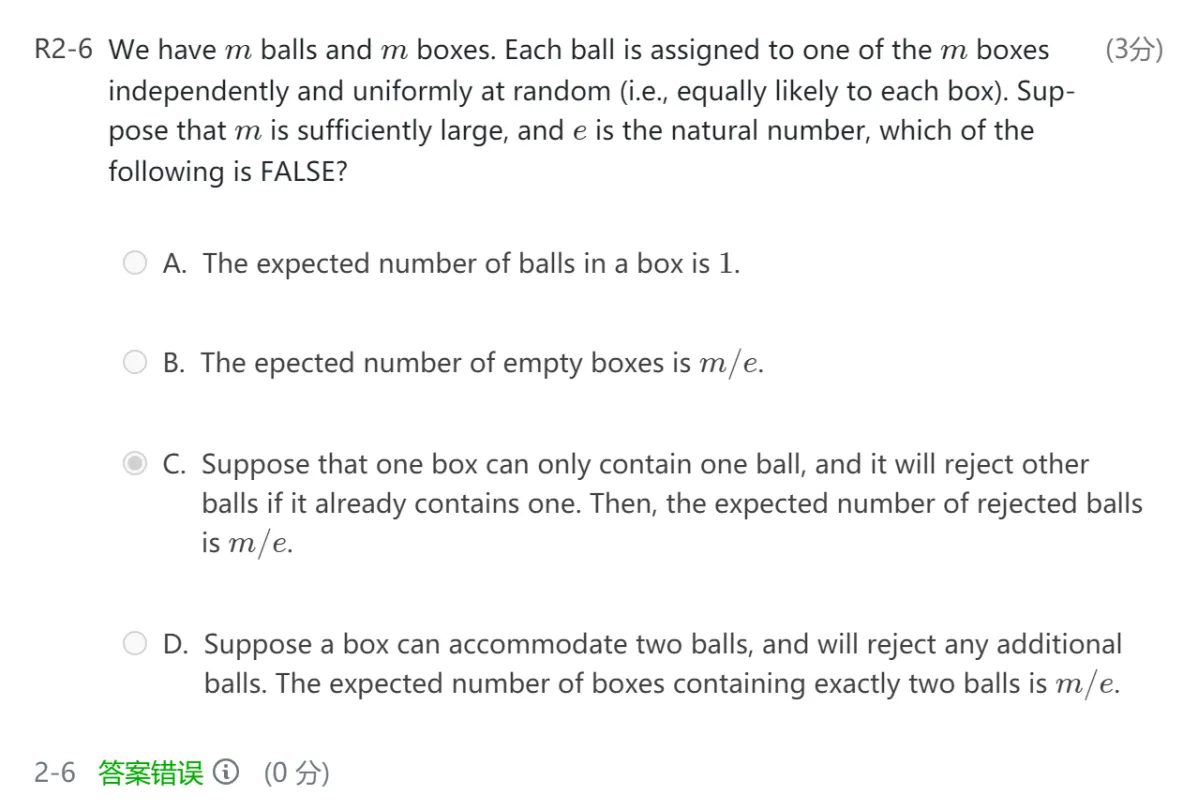
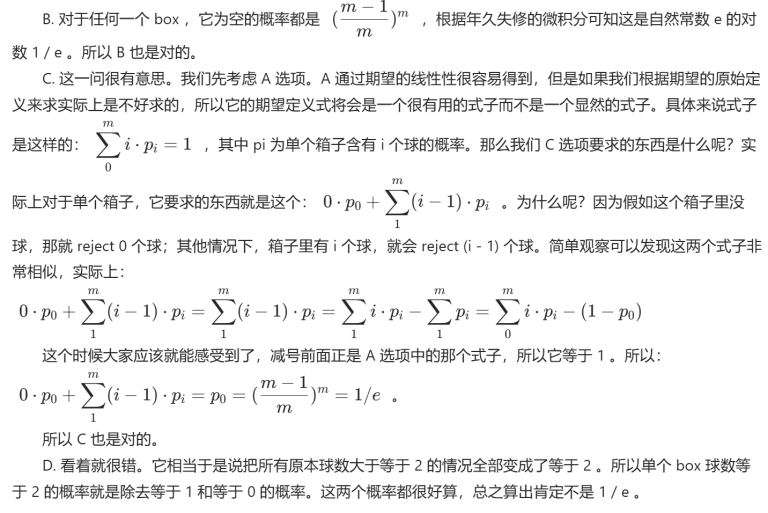
1. We have m balls and m boxes. Each ball is assigned to one of the m boxes independently and uniformly at random (i.e., equally likely to each box). Suppose that m is sufficiently large, and e is the natural number, which of the following is FALSE?

A.The expected number of balls in a box is 1.

B.Suppose a box can accommodate two balls, and will reject any additional balls. The expected number of boxes containing exactly one balls is m/e.

C.Suppose that one box can only contain one ball, and it will reject other balls if it already contains one. The expected number of boxes containing exactly one balls is m/e.

D.The epected number of empty boxes is m/e.

参考语雀解析（有点不同）：C 既然它是拒绝更多的球，最后问恰好有一个球的盒子的数量的期望，我们其实可以这样考虑：假设盒子没有限制条件，那么有球的盒子的数量的期望是多少？我们不是知道空盒的期望吗？减掉就好了。那么有球的盒子都会被变成恰好有一个球的盒子，因为多余的球被 rejected 了。另外，除了(1-1/x)x=e，还有(1-1/x)x-1=e

1. In typical applications of data structures, it is not a single operation that is performed, but rather a sequence of operations, and the relevant complexity measure is not the time taken by one operation but the total time of a sequence. Hence instead of imposing any explicit structural constraint, we allow the data structure to be in an arbitrary state, and we design the access and update algorithms to adjust the structure in a simple, uniform way, so that the efficiency of future operations is improved. We call such a data structure self-adjusting. For example skew heaps and splay trees are such kind of structures.

Which one of the following statements is FALSE about self-adjusting data structures?

A.Their access and update algorithms are easy to understand and to implement.

B.They need less space, since no balance information is kept.

C.Less local adjustments take place than in the corresponding balanced structures, especially during accesses.

D.In an amortized sense, ignoring constant factors, they can be at least as efficient as balanced structures.

C

7. Consider the bin packing problem which uses a minimum number of bins to accommodate a given list of items. Recall that Next Fit (NF) and First Fit (FF) are two simple approaches, whose (asymptotic) approximation ratios are 2 and 1.7, respectively. Now we focus on a special class I2 of instances in which only two distinct item sizes appear. Check which of the following statements is true by applying NF and FF on I2.

A.NF and FF both have improved approximation ratios.

B.Neither of NF or FF has an improved approximation ratio.

C.FF has an improved approximation ratio, while NF does not.

D.NF has an improved approximation ratio, while FF does not.

C

8.About Vertex Cover problem, which of the following statements is FALSE?

A.The time complexity of its verification algorithm is O(N3), where N refers to the number of nodes.

B.It is an NP-complete problem.

C.It is an NP problem.

D.It is polynomial-time reducible to Clique problem, but not vice versa.

D 相互规约

1. Splaying roughly halves the depth of most nodes on the access path

T

1. In a splay tree, for any non-root node X, its parent P and grandparent G (guranteed to exist), the correct operation to splay X to G is to rotate X upward twice.？

别管了、、、

1. To sort N numbers by external sorting using a k-way merge and a k-size heap, which statement is TRUE about the total comparison times T(N,k) and k?

A.T(N,k) has nothing to do with k.

B.T(N,k) is O(k2) for fixed N.

C.T(N,k) is O(klogk) for fixed N.

D.T(N,k) is O(k) for fixed N.

A 因为k路归并需要logkN轮，而因为有k路所以一个数的顺序的产生需要logk次比较，那一轮下来N个数就要Nlogk次比较，相乘得到NlogklogkN=NlogN

1. C

