R1-1 分数 2 作者 Yuchen Mao 单位 浙江大学

Let Σ be a set of at least two characters, each having a frequency. In any optimal prefix-free code for Σ, at least two characters have the same coding length.

(•) T ( ) F

评测结果 答案正确
得分 2 分

R1-2 分数 2 作者 Yuchen Mao 单位 浙江大学

Let $T(n)$ be the running time of quicksort on an input of size n. We already know that $T(n)$ is a random variable whose value depends on the random choices of quicksort, and that the expectation of $T(n)$ is $O(n \log n)$. Is the following statement true or false? The minimum possible value of $T(n)$ can be as small as $\Theta(n)$, and the maximum possible value can be as large as $\Theta(n^2)$.

( ) T (•) F

评测结果 答案正确
得分 2 分

R1-3 分数 2 作者 叶德仕 单位 浙江大学

If NP $\neq$ co-NP, then P $\neq$ NP.

(•) T ( ) F

评测结果 答案正确
得分 2 分

R1-4 分数 2 作者 叶德仕 单位 浙江大学

We have an approximation algorithm ALG for a minimization problem X. For any given $\varepsilon > 0$, the approximation ratio of ALG is $1 + \varepsilon$ and the running time is $O(3^{1/\varepsilon} n^2)$, where n is the input size of the problem X. Then the algorithm ALG is an FPTAS.

( ) T (•) F

评测结果 答案正确
得分 2 分

R1-5 分数 2 作者 刘一辰 单位 浙江大学

Insert 2,9,5,6,1,8, and 3 one by one into an initially empty AVL tree. Then the postorder traversal sequence of the resulting tree must be {1,3,2,6,9,8,5}.

(•) T ( ) F

评测结果 答案正确
得分 2 分

R1-6 分数 2 作者 叶德仕 单位 浙江大学

For a leftist heap, the NPL of each node is 1 greater than the NPL of its right child.

(•) T ( ) F

评测结果 答案正确
得分 2 分

R1-7 分数 2 作者 卜佳俊 单位 浙江大学

In the dynamic indexing situation, the auxiliary index is usually updated when a new document comes to the document collection.

( ) T (•) F

评测结果 答案错误
得分 0 分

R1-8 分数 2 作者 卜佳俊 单位 浙江大学

There are two statements about Local Search:

- For any local search algorithm, searching a better solution in the neighborhood can always be done in polynomial time.
- For any local search algorithm, it takes polynomial time to find the local minimum.

Both of the statements above are correct.

( ) T (•) F

评测结果 答案正确
得分 2 分

R1-9 分数 2 作者 陈越 单位 浙江大学

Priority CRCW allows concurrent access for both reads and writes, while the processor with the smallest number has the highest priotity.

- (●) T
- ( ) F

评测结果 答案正确

得分 2 分

R1-10 分数 2 作者 陈越 单位 浙江大学

The time taken to check if a partial solution satisfies the restrictions is relatively hard (that is, no definite polynomial-time method works) to estimate during backtracking.

- (●) T
- ( ) F

评测结果 答案错误

得分 0 分

R1-11 分数 2 作者 陈昊 单位 浙江大学

In amortized analysis, the change in potential should be negative for low-cost operations and positive for high-cost operations.

- ( ) T
- (●) F

评测结果 答案正确

得分 2 分

R1-12 分数 2 作者 陈昊 单位 浙江大学

Given 4000 runs and 10 tapes. If simple k-way merge is used, the minimum number of passes required is 5 (runs generation pass is not counted).

- ( ) T
- (●) F

评测结果 答案正确

得分 2 分

R1-13 分数 2 作者 丁尧相 单位 浙江大学

The recurrent equation $T(n) = nT(n/2) + n$ can be solved by the master theorem.

- ( ) T
- (●) F

评测结果 答案正确

得分 2 分

R2-1 分数 3 作者 Yuchen Mao 单位 浙江大学

Consider the Knapsack Problem. How many of the following statements are true?

- If all the items have the same weight, then we can obtain an optimal solution by selecting items in decreasing order of profits.
- If all the items have the same profit, then we can obtain an optimal solution by selecting items in increasing order of weights.
- If all the items have the same efficiency (that is, profit-to-weight ratio), then we can obtain an optimal solution by selecting items in increasing order of weights.

- ( ) A. 0
- (●) B. 1
- ( ) C. 2
- ( ) D. 3

评测结果 答案错误

得分 0 分

R2-2 分数 3 作者 Yuchen Mao 单位 浙江大学

Consider eight characters with the following frequencies. (We normalize the frequencies so that they sum to 1.)

Symbol	Frequency
A	0.11
B	0.11
C	0.11
D	0.11
E	0.14
F	0.14
G	0.14
H	0.14

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

- ( ) A. 2.87
- ( ) B. 2.94
- ( ) C. 2.97
- (●) D. 3

评测结果 答案正确

得分 3 分

R2-3 分数 3

作者 Yuchen Mao 单位 浙江大学

Consider the hiring problem. Assume that the n candidates arrive in random order, and that no two candidates have the same performance. Let k be some even number (偶数) in the range $[1, n]$. We use the following algorithm:

```
interview the first k candidates, but hire none of them
for candidates i = k+1 to n
    if candidate i is better than at least half of the first k candidates
        hire candidate i
```

It is easy to see that the above algorithm may hire more than one candidates. How many candidates will be hired in expectation?

- A. $\frac{(n-k)(k+2)}{2k+2}$
- B. $\frac{(n-k)}{k+1}$
- C. $\frac{2(n-k)}{k+1}$
- D. $\frac{(n-k)\ln(n-k)}{k+1}$

评测结果 答案正确

得分 3 分

R2-4 分数 3

作者 叶德仕 单位 浙江大学

A **Hamiltonian Cycle** (or Circuit) in a graph is a cycle that visits every vertex exactly once and returns to the starting vertex. Finding a Hamiltonian Cycle is a well-known NP-complete problem. Let's consider the following two new variants of the Hamiltonian problems:

P1) **Long Cycle problem**: A Long Cycle in a graph G is a cycle that goes through at least half of the vertices of G.

P2) **Hamiltonian Path problem**: A graph G has a Hamiltonian path from s to t if there is an s to t path that visits all of the vertices of G exactly once.

Which of the following statements is correct?

- A. Only the Long Cycle problem is NP-Complete.
- B. Only the Hamiltonian Path problem is NP-complete.
- C. Both the Long Cycle problem and the Hamiltonian Path problem are NP-complete.
- D. None of the Long Cycle problem and the Hamiltonian Path problem are NP-complete.

评测结果 答案错误

得分 0 分

R2-5 分数 3

作者 叶德仕 单位 浙江大学

In the **Activity Selection Problem**, we are given a set of activities $S = \{a_1, a_2, \ldots, a_n\}$ that wish to use a resource. Each a_i takes place during a time interval $[s_i, f_i)$. Activities a_i and a_j are compatible if $s_i \geq f_j$ or $s_j \geq f_i$ (i.e. their time intervals do not overlap). Our goal is to select a maximum-size subset of mutually compatible activities.

We propose a greedy rule "Shortest-First" that select the interval which is the **shortest** (but not overlapping the already chosen intervals). What is the **approximation ratio** of the greedy algorithm for the activity selection problem?

- A. 3/2
- B. 2
- C. 3
- D. no constant approximation ratio.

评测结果 答案正确

得分 3 分

R2-6 分数 3

作者 刘一辰 单位 浙江大学

Insert 2,9,5,6,1,8, and 3 one by one into an initially empty Splay tree. Which one of the following statements is TRUE about the resulting tree?

- A. 5 and 9 are siblings
- B. 8 is the parent of 5
- C. 1 and 8 are siblings
- D. 3 is the parent of 2

评测结果 答案正确

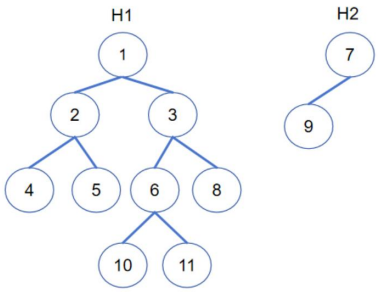
得分 3 分

R2-7 分数 3

作者 叶德仕 单位 浙江大学

Merge the two leftist heaps in the figure. How many of the following statements is/are FALSE ?

- 8 and 9 are siblings
- 6 and 7 are siblings
- 1 and 3 have the differernt NPL
- along the left path from the root, we have 1, 3, 6,10



- A. 1
- B. 2
- C. 3
- D. 4

评测结果 答案错误

得分 0 分

R2-8 分数 3

作者 卜佳俊 单位 浙江大学

There are 100000 documents in the document collection. The statistic data for one query is shown in the following table.

	Relevant	Irrelevant
Retrieved	10000	30000
Not Retrieved	40000	20000

Which one of the following statements about the statistics is True?

- A. The Recall is 0.25.
- B. The Precision is 0.1.
- C. The Precision is 0.2.
- D. None of the other statements is correct.

评测结果 答案正确

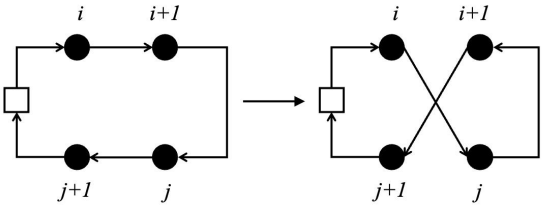
得分 3 分

R2-9 分数 3

作者 卜佳俊 单位 浙江大学

The Traveling Salesman Problem (TSP) is a classic problem: given a set of cities and the distances between each pair of cities, the goal of TSP is to find the shortest cycle that visits each city exactly once and returns to the starting city.

The 2-opt method is a local search algorithm used to solve the TSP. Specifically, the searching strategy is as follows:
(1) Starting at a feasible solution to the TSP (a valid cycle which visits each city exactly once and returns to the starting city).
(2) In each iteration, the current cycle is denoted as P. The neighboring solution P' is obtained by a 2-opt move. The 2-opt move contains the following steps:
  i) Choosing two non-adjacent edges $(i, i+1)$ and $(j, j+1)$ from the cycle;
  ii) Removing these two edges, and replacing them with new edges (i, j) and $(i+1, j+1)$, forming a new cycle P'.
If the new cycle obtained by the 2-opt move is shorter than the current cycle, updating the current cycle as the new cycle.
(3) Repeating the step (2) until no further improvement can be made by applying 2-opt moves.
Here is an illustration of a 2-opt move.



Which of the following statements is true?

- A. The TSP is NP-hard, so the 2-opt iteration will not terminate.
- B. When no 2-opt move can improve the current cycle, the algorithm reaches the global optimum.
- C. When applying the 2-opt method, different initial cycles may lead to different local optimums.
- D. After each 2-opt move, the solution is improved by at least a constant ratio.

评测结果 答案正确

得分 3 分

R2-10 分数 3

作者 陈越 单位 浙江大学

In the theory of parallel algorithms using PRAM, which one of the following statements is FALSE?

- A. The work complexity of a parallel algorithm is never less than the best sequential time for the same problem.
- B. Parallel computing makes debugging simpler since the data processing is more visible.
- C. The run time of a parallel algorithm depends not only on the size of the problem, but also on the number of available processors.
- D. To measure the performance of a parallel algorithm, it is asymptotically equivalent to consider the following two methods: (1) $W(n)$ operations and $T(n)$ time; and (2) $P(n) = W(n)/T(n)$ processors and $T(n)$ time.

评测结果 答案正确

得分 3 分

R2-11 分数 3

作者 陈越 单位 浙江大学

Consider two parallel algorithms for the same problem, with a total of $W_i(n)$ operations in $T_i(n)$ time ($i = 1, 2$). If $W_1(n) = O(n), T_1(n) = O(n)$, and $W_2(n) = O(n \log n), T_2(n) = O(\log n)$. Then:

- A. the first algorithm is more efficient than the second algorithm
- B. the first algorithm is less efficient than the second algorithm
- C. the performances of the two algorithms are asymptotically equivalent
- D. none of the other three options are correct

评测结果 答案正确

得分 3 分

R2-12 分数 3

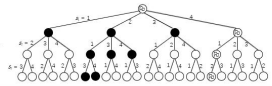
作者 陈越 单位 浙江大学

Suppose we have 4 types of fruit for 3 monkeys. Each monkey has its favorite types of fruit, and the matrix $T = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$ gives the corresponding relations between the monkeys and fruits -- that is, $t_{ij} = 1$ means the $i$the monkey loves the jth type of fruit, or 0 means the opposite. Use backtracking to find a way to send each monkey its favorite fruit, with the following restrictions:

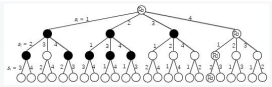
1. one monkey may get only one type of fruit; and
2. one type of fruit can be sent to at most one monkey.

Which one of the following game trees corresponds to a backtracking process?
Note: the black nodes are the ones being pruned, and the white ones are never visited. The nodes with a flag correspond to the solution.

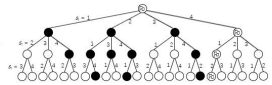
A.



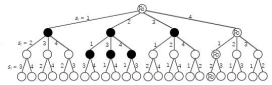
B.



C.



D.



评测结果 答案正确

得分 3 分

R2-13 分数 3

作者 陈昊 单位 浙江大学

There are four basic operations on red-black trees that perform **structural modifications**: node insertions, node deletions, rotations, and color changes. We shall prove that any sequence of m RB-INSERT and RB-DELETE operations on an initially empty red-black tree causes $O(m)$ structural modifications in the worst case. **We count the structural modifications in each step (e.g. Case 1 in RB-DELETION) as one unit operation (cost = 1)**.

We define the weight of each node based on its state, and the potential of the Red-Black Tree T is represented by the following function:

$\Phi(T) = \sum_{x \in T} g(x)$

where $g(x)$ is calculated for all nodes $x \in T$ of the Red-Black Tree.

We define the weight of a **red node** x as $g(x) = 0$.

For black nodes, which of the following definitions work?

A.
- $g(x) = 1$: If the black node has **no red children**.
- $g(x) = 0$: If the black node has **one red child**.
- $g(x) = 2$: If the black node has **two red children**.

B.
- $g(x) = 1$: If the black node has **no red children** or **one red child**.
- $g(x) = 2$: If the black node has **two red children**.

C.
- $g(x) = 0$: If the black node has **no red children**.
- $g(x) = 1$: If the black node has **one red child**.
- $g(x) = 2$: If the black node has **two red children**.

D.
- $g(x) = 1$: If the black node has **no red children**.
- $g(x) = 2$: If the black node has **one red child**.
- $g(x) = 0$: If the black node has **two red children**.

评测结果 答案错误

得分 0 分

R2-14 分数 3

作者 陈昊 单位 浙江大学

Suppose that replacement selection is applied in **external sorting** to generate longer runs with a priority queue of size 4. Given the sequence of numbers {51, 94, 37, 92, 14, 63, 15, 99, 48, 56, 23, 60, 31, 17, 43, 8, 90, 166, 100}. How many of the following states are **TRUE**?

- 3 runs will be generated
- 14 is in the first runs
- The length of the longest run is 9
- 166 is in the last runs

A. 1 B. 2 C. 3 D. 4

评测结果 答案正确

得分 3 分

R2-15 分数 3

作者 丁尧相 单位 浙江大学

If a binomial queue consists of 3 trees, which of the following numbers can NOT be its number of nodes?

A. 6 B. 26 C. 28 D. 1041

评测结果 答案正确

得分 3 分

R2-16 分数 3

作者 丁尧相 单位 浙江大学

A key advantage of binomial queue is that its amortized time cost of insertion is $O(1)$. To prove this argument, a key observation is that if inserting one new key in the queue takes time cost c, it will lead to increasing m trees in the queue, where m equals to

A. $-c$ B. $2-c$ C. $c-2$ D. c

评测结果 答案正确

得分 3 分

R2-17 分数 3

作者 丁尧相 单位 浙江大学

What is the tightest solution to the recurrent function $T(n)=2T(n/2)+n\log n$?

A. $T(n)=O(n\log n)$ B. $T(n)=O(n\log^2 n)$ C. $T(n)=O(n^2)$ D. $T(n)=O(n^2\log n)$

评测结果 答案正确

得分 3 分

R2-18 分数 3

作者 丁尧相 单位 浙江大学

How many of the following arguments are TRUE?

1. In comparison to dynamic programming, divide-and-conquer algorithms are usually more suitable for problems with highly overlapping sub-problems.
2. In the divide-and-conquer solution to the "closest pair of points in the plane" problem, sorting the points in both X and Y axes at the beginning of the whole algorithm is essential to achieve the $O(n\log n)$ time complexity.
3. Consider the view point of treating function $T(n)=aT(n/b)+n^c$ as a recursion tree. If $T(n)=O(n^c)$, then the time cost is dominated by the root of the tree.

A. 0 B. 1 C. 2 D. 3

评测结果 答案错误

得分 0 分

R2-19 分数 3

作者 杨洋 单位 浙江大学

Suppose a red-black tree T contains $N(N\geq 2)$ internal nodes, we denote the height of T as $h(T)$ and the black-height of T as $bh(T)$. Which of the following statements must be FALSE (assume the height of an empty tree is 0)?

A. $h(T)\leq 2bh(T)$ B. $h(T)=3\lceil\log_2(N+1)\rceil$

C. The number of leaf nodes (NIL) in T is $2N-1$ D. The number of leaf nodes (NIL) in T is $3N-5$

评测结果 答案正确

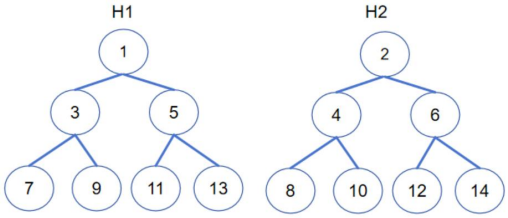
得分 3 分

R2-20 分数 3

作者 叶德仕 单位 浙江大学

Merge the two skew heaps in the following figure. How many of the following statements is/are FALSE?

- the null path length of 1 is 2
- 12 is the left child of 6
- the depths of 4 and 9 are the same



A. 0 B. 1 C. 2 D. 3

评测结果 答案正确

得分 3 分

R5-1 分数 6 Fill in B+ Tree

作者 杨洋 单位 浙江大学

Given a B+ Tree of order `odr`, please calculate the maximum number of keys that can be inserted into the current tree `root` without causing any split operation.

```
typedef struct BpTreeNode BpTreeNode;
struct BpTreeNode {
    BpTreeNode** childrens; /* Pointers to childrens. This field is not used by leaf nodes. */
    ElementType* keys;
    BpTreeNode* parent;
    bool isLeaf; /* 1 if this node is a leaf, or 0 if not */
    int numKeys; /* This field is used to keep track of the number of valid keys.*/
};
int odr;

int Solve(BpTreeNode * const root){
    BpTreeNode * node = root;
    if (node->isLeaf) {
        return odr - node->numKeys ; // 3分
    }
    int ans = 0;
    for(int i = 0; i < node->numKeys + 1 ; i++) // 3分
        ans += Solve(node->childrens[i]);
    return ans;
}
```

评测结果 答案正确

得分 6 分

R6-1 Knapsack 分数 8

全屏浏览 切换布局

作者 王灿 单位 浙江大学

Given n items, where the i-th one has weight w_i and value v_i. We only have a knapsack with capacity W which might not be large enough to hold all the items. Fortunately, we can use magic — for item i, each time the magic can reduce its weight by 1 for the cost of c_i. On the other hand, the weight of any item must always be positive — meaning that we cannot use magic to reduce the weight to 0 or less. We can use the magic any number of times. The profit of packing item i into the knapsack is $v_i - k_i \times c_i$ where k_i is the number of times we apply the magic on item i.

The question is: what is the maximum profit we can get?

It is guaranteed that $1 \le n \le 200, 1 \le W \le 200, 1 \le w_i \le 200, 1 \le v_i \le 10^6, 1 \le c_i \le 10^6$.

Function Interface:

```
1 int knapsack(int n, int W, int w[], int v[], int c[]);
```

where `n` represents the number of items. `W` represents the capacity of the knapsack. The arrays `w[0…n-1]`,`v[0…n-1]`,`c[0…n-1]` are the weights, values, and costs of the items.

```
#include <stdio.h>

int knapsack(int n, int W, int w[], int v[], int c[]);
const int maxn = 200;

int main()
{
    int n, W;
    int v[maxn], w[maxn], c[maxn];
```

```
    scanf(“%d %d”, &n, &W);
    for(int i = 0; i < n; i++)
        scanf(“%d%d%d”, &w[i], &v[i], &c[i]);
    printf(“%d\n”, knapsack(n, W, w, v, c));

    return 0;

}
/* Fill your program here*/
```

Sample Input:

```
4 10
8 10 2
12 15 3
5 9 2
3 8 1
```

Sample Output:

```
17
```

代码长度限制	16 KB
时间限制	400 ms
内存限制	64 MB