ADS2021Midterm-zgc(20min)

开始时间	2022/04/15 17:00:00	**结束时间**	2022/04/29 12:00:00	**答题时长**	20分钟
答卷类型	标准答案	**总分**	100		

判断题

得分：暂无　总分：45

1-1 When measuring the relevancy of the answer set, if the precision is high but the recall is low, it means that most of the relevant documents are retrieved, but too many irrelevant documents are returned as well. (5分)

○ T ◉ F

1-2 In the problem of N Queens, since the game tree contains $N!$ leaves, the space complexity for solving the problem is $\Omega(N!)$. (5分)

○ T ◉ F

1-3 Making N insertions into an initally empty binomial queue takes $\Theta(NlogN)$ time in the worst case. (5分)

○ T ◉ F

1-4 Amortized analysis is a worst-case analysis of a sequence of operations, so that a tighter bound on the overall or average cost per operation in the sequence is obtained. (5分)

◉ T ○ F

1-5 Givien two $n \times n$ matrices A and B, the time complexity of the simple matrix multiplication $C = A \cdot B$ is $O(n^3)$. Now let's consider the following Divide and Conquer idea: (5分)

Divide each matrix into four $\frac{n}{2} \times \frac{n}{2}$ submatrics as follows:

$$\begin{bmatrix} C_1 & C_2 \\ C_3 & C_4 \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \cdot \begin{bmatrix} B_1 & B_2 \\ B_3 & B_4 \end{bmatrix}$$

We recursively calculate each block of C as $C_1 = A_1 \cdot B_1 + A_2 \cdot B_3$ and so on. This can reduce the time complexity of the simple calculation.

○ T ◉ F

1-6 A 2-3 tree with 12 leaves may have at most 10 nonleaf nodes. (5分)

◉ T ○ F

1-7 Finding the minimum key from a splay tree will result in a tree with its root having no left subtree. (5分)

◉ T ○ F

1-8 An ordered stack is a data structure that stores a sequence of items and supports the following operations. (5分)

- `OrderedPush(x)` removes all items smaller than `x` from the beginning of the sequence and then adds `x` to the beginning of the sequence.
- `Pop` deletes and returns the first item in the sequence (or `Null` if the sequence is empty).

Suppose we implement an ordered stack with a simple linked list, using the previous `OrderedPush` and `Pop` algorithms.

It is proved that if we start with an empty data structure, the amortized cost of each `OrderedPush` and `Pop` operations is $O(1)$.

◉ T ○ F

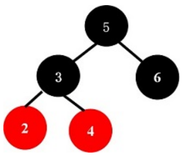
1-9 Insert 1, 2, 3, 4, 5, and 6 one by one into an initially empty AVL tree. Then the preorder traversal sequence of the resulting tree must be {4, 2, 1, 3, 5, 6}. (5分)

◉ T ○ F

单选题

得分：暂无　总分：40

2-1 After inserting 1 into the red-black tree given in the figure, which node(s) will have its/their color(s) changed? (10分)



○ A. only 2
○ B. both 2 and 4
◉ C. 2, 3, and 4
○ D. both 3 and 6

2-2 To solve the optimal binary search tree problem, we have the recursive equation $c_{ij} = \min_{i<l\leq j}\{w_{ij} + c_{i,l-1} + c_{l+1,j}\}$. To solve this equation in an iterative way, we must fill up a table as follows: (10分)

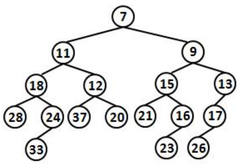
○ A.
```
for i= 1 to n-1 do;
  for j= i to n do;
    for l= i to j do
```

○ B.
```
for j= 1 to n-1 do;
  for i= 1 to j do;
    for l= i to j do
```

◉ C.
```
for k= 1 to n-1 do;
  for i= 1 to n-k do;
    set j = i+k;
    for l= i to j do
```

○ D.
```
for k= 1 to n-1 do;
  for i= 1 to n do;
    set j = i+k;
    for l= i to j do
```

2-3 Delete the minimum number from the given leftist heap. Which one of the following statements is TRUE? (10分)



○ A. 9 is NOT the root
○ B. 24 is the left child of 18
◉ C. 13 is the left child of 12
○ D. 18 is the right child of 11

2-4 When solving a problem with input size N by divide and conquer, if at each step, the problem is divided into 4 sub-problems of size $\sqrt{N}$, and they are conquered in $O(logN)$. Which one of the following is the closest to the overall time complexity $T(N)$? Suppose that $T(2) = O(1)$ and all related root values are integers. (10分)

○ A. $O(logN)$
◉ B. $O(log^2N)$
○ C. $O(N)$
○ D. $O(loglogN)$

程序填空题

得分：暂无　总分：15

5-1 The functions `IsRBT` is to check if a given binary search tree `T` is a red-black tree. Return `true` if `T` is, or `false` if not.

The red-black tree structure is defined as the following:

```
typedef enum { red, black } colors;
typedef struct RBNode *PtrToRBNode;
struct RBNode{
    int Data;
    PtrToRBNode Left, Right, Parent;
    int BlackHeight;
    colors Color;
};
typedef PtrToRBNode RBTree;
```

Please fill in the blanks.

```
bool IsRBT( RBTree T )
{
    int LeftBH, RightBH;
    if ( !T ) return true;
    if ( T->Color == black ) T->BlackHeight = 1;
    else {
         if ( T->Left && [(T->Left->Color == red)] (5分)) return false;
         if ( T->Right && (T->Right->Color == red) ) return false;
    }
    if ( !T->Left && !T->Right ) return true;
    if ( [IsRBT( T->Left ) && IsRBT( T->Right )] (5分)) {
       if ( T->Left ) LeftBH = T->Left->BlackHeight;
       else LeftBH = 0;
       if ( T->Right ) RightBH = T->Right->BlackHeight;
       else RightBH = 0;
       if ( LeftBH == RightBH ) {
           [T->BlackHeight += LeftBH] (5分);
           return true;
       }
       else return false;
    }
    else return false;
}
```