2020–2021学年秋冬学期

《数据结构基础》课程期中考试试卷（何钦铭班）

一、判断题（10分，共5题，每题2分）

1-1 If the pre-order traversal sequence of a binary tree is `ABC`, then `CAB` **cannot** be its in-order traversal sequence.

1-2 To find `63` from a binary search tree, one possible searching sequence is `{39, 101, 25, 80, 70, 59, 63}`.

1-3 Given two sorted lists `L1` and `L2`, the fastest algorithm for computing `L1` ∪ `L2` has time complexity $\Theta(N)$.

1-4 2^N and N^N have the same speed of growth.

1-5 Given that the pushing sequence of a stack is `{1, 2, ⋯, n}` and popping sequence is $\{x_1, x_2, \cdots, x_n\}$. If $x_2 = n$, we can obtain 2 different possible popping sequences.

二、选择题（70分，共14题，每题5分）

2-1 If `A` and `B` are both leaf nodes in a binary tree, then which of the following is TRUE?

A. There exists a binary tree with pre-order traversal sequence `...A...B...` and post-order traversal sequence `...B...A...`.

B. There exists a binary tree with pre-order traversal sequence `...A...B...` and in-order traversal sequence `...B...A...`.

C. None of the above.

D. There exists a binary tree with in-order traversal sequence `...A...B...` and post-order traversal sequence `...B...A...`.

2-2 The array representation of a disjoint set containing numbers 0 to 8 is given by `{1, -4, 1, 1, -3, 4, 4, 8, -2}`. Then to union the two sets which contain 6 and 8 (with union-by-size), the index of the resulting root and the value stored at the root are:

A. `1` and `-6` B. `8` and `-6` C. `8` and `-5` D. `4` and `-5`

2-3 A full tree of degree 4 is a tree in which every node other than the leaves has 4 children. How many leaves does a full tree of degree 4 have if it has 257 nodes?

A. 64 B. 193 C. 63 D. 194

2-4 Given an empty stack `S` and an empty queue `Q`. Push elements `{1, 2, 3, 4, 5, 6, 7}` one by one onto `S`. If each element that is popped from `S` is enqueued onto `Q` immediately, and if the dequeue sequence is `{4, 5, 7, 6, 3, 2, 1}`, then the minimum size of `S` must be:

A. 5 B. 3 C. 4 D. 2

2-5 Insert `{28, 15, 42, 18, 22, 5, 40}` one by one into an initially empty binary search tree. The post-order traversal sequence of the resulting tree is:

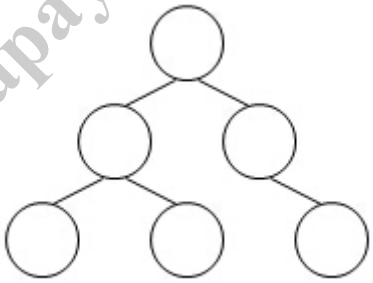
A. `5, 22, 15, 40, 18, 42, 28` B. `5, 15, 18, 22, 40, 42, 28`

C. `5, 22, 18, 15, 40, 42, 28` D. `28, 22, 18, 42, 40, 15, 5`

2-6 If the pushing sequence of a stack is `{1, 2, 3, ..., N}` and the popping sequence is $\{p_1, p_2, p_3, \ldots, p_N\}$. Given that $p_1 = N$, then p_i must be:

A. i B. $n - i$ C. $n - i + 1$ D. cannot be determined

2-7 Given the structure of a binary search tree (as shown in the figure), which one of the following insertion sequences is impossible?



A. `85 89 95 56 75 18` B. `85 56 89 95 18 75`

C. `85 56 75 89 18 95` D. `85 89 75 56 18 95`

2-8 Suppose that all the leaf nodes of a given non-empty complete binary tree T are on the same level, and every non-leaf node has two children. If there are k leaf nodes in T, then T must have ____ nodes in total.

A. k^2 B. $2k - 1$ C. $2k$ D. $2^k - 1$

2-9 Using the linear algorithm to build a min-heap from the sequence `{16, 28, 14, 13, 7, 6}`, and then DeleteMin. Which one of the following statements is false?

A. `13` and `14` are siblings.

B. `7` is the root.

C. `28` is the left child of `13`.

D. `13` is the parent of `16`.

2-10 Given a stack `S` and a queue `Q`, where `S` is initialized to be empty and `{1, 2, 3, 4, 5, 6}` are in `Q` (with `1` stored at the front end). If only the following three operations are allowed: (1) delete and print an element from `Q`; (2) delete an element from `Q` and push it onto `S`; (3) pop and print an element from `S`, which of the following output sequences is NOT possible?

A. `6, 5, 4, 3, 2, 1` B. `3, 4, 5, 6, 1, 2`

C. `1, 2, 5, 6, 4, 3` D. `2, 3, 4, 5, 6, 1`

2-11 Let n be a non-negative integer representing the size of input. The time complexity of the following piece of code is:

```
x = 0;
while ( n >= (x+1)*(x+1) )
    x = x+1;
```

A. $O(n)$ B. $O(n^{1/2})$ C. $O(n^2)$ D. $O(\log n)$

2-12 If a stack is used to convert the infix expression `a+b*c+(d*e+f)*g` into a postfix expression, what will be in the stack (listing from the bottom up) when `f` is read?

A. `+(+` B. `abcde` C. `++(+` D. `+(*+`

2-13 For two linear lists `La` and `Lb`, to link the tail of `La` with the head of `Lb`, with which one of the following data structures that we can take $O(1)$ time and minimize the extra space?

A. singly linked circular list with a tail pointer

B. singly linked circular list

C. singly linked list

D. doubly linked circular list with a dummy head node

2-14 Insert `{6, 9, 12, 3, 4, 8}` one by one into an initially empty binary search tree. The post-order traversal sequence of the resulting tree is:

A. `3, 4, 9, 8, 12, 6` B. `3, 4, 6, 8, 12, 9`

C. `4, 3, 8, 12, 9, 6` D. `4, 3, 6, 8, 12, 9`

三、程序填空题（20分，共4个空，每空5分）

5-1 The function is to find the `K`-th smallest element in a list `A` of `N` elements. The function `BuildMaxHeap(H, K)` is to arrange elements `H[1]` ... `H[K]` into a max-heap. Please complete the following program.

```
ElementType FindKthSmallest ( int A[], int N, int K )
{   /* it is assumed that K<=N */
    ElementType *H;
    int i, next, child;

    H = (ElementType *)malloc((K+1) * sizeof(ElementType));
    for (i = 1; i <= K; i++)
        H[i] = A[i-1];
    BuildMaxHeap(H, K);

    for (next = K; next < N; next++)
    {
        H[0] = A[next];
        if (H[0] < H[1]) {
            for (i = 1; i * 2 <= K; i = child)
            {
                child = i*2;
                if (child != K && _________ )
                    child++;
                if ( __________ )
                    H[i] = H[child];
                else
                    break;
            }
            H[i] = H[0];
        }
    }
    return H[1];
}
```

5-2 The function is to return the reverse linked list of `L`, with a dummy header.

```
List Reverse( List L )
{
    Position Old_head, New_head, Temp;
    New_head = NULL;
    Old_head = L->Next;

    while (Old_head) {
        Temp = Old_head->Next;
        __________;
        New_head = Old_head;
        Old_head = Temp;
    }
    __________;
    return L;
}
```