Sichuan University Final Examination

(Closed Book)

(2016-2017 Academic Year 1st Semester)

课程号 Course Number: 311116030 课程名称: Data Structure and Algorithm
任课教师 Lecturer: Yingke Che 考试时间 Time Period: 13:50 – 15:50 (Jan-4-2017)
学生姓名 Name: 学号 Student ID: 成绩 Total Mark:

考 生 承 诺

Student Commitment

我已认真阅读并知晓《四川大学考场规则》和《四川大学本科学生考试违纪作弊处分规定（修订）》，郑重承诺：

I have read and comprehended the "Regulations of Sichuan University on Examinations". I give my commitments as follows:

1、已按要求将考试禁止携带的文具用品或与考试有关的物品放置在指定地点；

1. I have put prohibited stationary and exam-related items at designated area as required.

2、不带手机进入考场；

2. I have not brought cell phone to the examination room.

3、考试期间遵守以上两项规定，若有违规行为，同意按照有关条款接受处理。

3. During the examination, I will comply with the above two provisions. If there is any violation, I agree to accept the punishments in accordance with the relevant provisions.

考生签名：
Signature:

I. Multiply choice (20 points)

1	2	3	4	5	6	7	8	9	10

1. What is the minimum number of nodes in a complete binary tree with depth 3?

( )

A. 3； B. 4； C. 8； D. 11；

2. The correct operation of inserting a node ‘s’ after node ‘p’ in a single linked list is:

A. p. next=p B. s. next=p. next, p. next=s

C. p. next=s； s. next=p. next D. p. next=s

3. Which one of following is quite right? Stacks and queues are both ( )

① linear structure with array

② linear structure with linked list

③ linear structure with array or linked list

④ linear structure whose insertion and deletions are restricted to the end

A. ①; B. ①④; C. ②④; D. ③④;

4. The addresses which store linked list .

A. must be sequential B. must be partly sequential

C. must be no sequential D. can be sequential or discontiguous

5. Given a stack, by the operation ( ), input ABC can be transform to CBA

A．push，pop，push，pop，push，pop

B．push，push，push，pop，pop，pop

C．push，push，pop，pop，push，pop

D．push，pop，push，push，pop，pop

6. Assume the preorder of binary tree T is ABEGFCDH, the inorder is EGBFADHC, then the postorder will be:

A. GEFBHDCA B. EGFBDHCA

C. GEFBDHCA D. GEBFDHCA

7. What is the **worst-case** time for binary search finding a single item in an array?

A. O(1) B. O(log(N)) C. O(N) D. O(N^2)

8. What is the best definition of a collision in a hash table? ( )

A. Two entries are identical except for their keys.

B. Two entries with different data have the exact same key.

C. Two entries with different keys have the same exact hash value.

D. Two entries with the exact same key have different hash values.

9. Consider the node of a complete binary tree whose value is stored in data[i] for an

array implementation. If this node has a right child, where will the right child's value be stored? ( )

A. data[i+1] B. data[i+2] C. data[2*i + 1] D. data[2*i + 2]

10. The correct traversal to use on a BST to visit the nodes in sorted order is( )

A. Preorder traversal; B. Inorder traversal;

C. Postorder traversal D. None of above

II. Fill blanks and True or False (10 points)

1	2	3	4		5

1. Given a linked list with n elements, the average time complexity for reading the i-th element is ( ).

2. Given a binary search tree with n nodes, we can always find an element with time complexity O($log \ \ n$). ( ) (True or False)

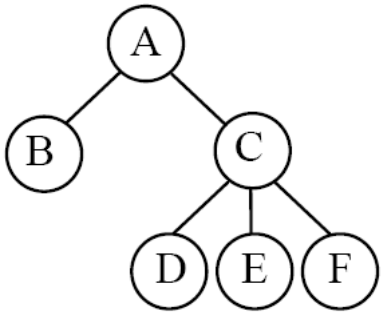
3. The big-O notation of arithmetic expression $5n^4 + 10nlog_2 n + 100nlog_2(n^5)$ is (_____)

4. Every complete binary tree is full. ( ) (True or False), and every full binary tree is complete. ( ) (True or False).

5. For binary trees, every node has at least two children. ( ) (True or False).

III. Analysis. Provide details as much as you can.

1. Write down the pre-order, post-order and in-order traversal of the tree below.



2. Convert (A + B) * C - (D - E) * (F + G) into a postfix expression, and convert ab+c/-de* into a infix expression. Provide detailed steps.

3. Give a sequence of numbers: 77, 22, 9, 68, 16, 34, 13, please sort it with two different sorting algorithms: selection sort and quick sort. Draw the result after each iteration/pass for each algorithm. For quick sort, indicate the pivot number for each pass.

4. Draw the binary **min** heap that results from inserting: 77, 22, 9, 68, 16, 34, 13, 8 in that order into an initially empty binary min heap. Draw the binary **min** heap that results from doing 2 deletions on the heap you created.

5. Given input {10, 3, 6, 16, 17, 19}, and a hash table with size 11, show the resulting:
(a) Separate chaining hash table with hash function h(x)=x mod 11
(b) hash table with second hash function h2=7-(x mod 7)

IV. Program. Provide details if necessary.

1. Given two lists $A=(a_1,a_2,\cdots,a_m)$, $B=(b_1,b_2,\cdots,b_n)$, please finish the following function to merge A and B into list C by the following rules,

$C=(a_1,b_1,\cdots,a_m,b_m,b_{m+1},\cdots,b_n)$ **, if** $m\le n$

$C=(a_1,b_1,\cdots,a_n,b_n,a_{n+1},\cdots,a_m)$ **, if** $m>n$

Lists A, B, and C are linked lists. The length of lists A and B are not given in advance.

```
LinkList mergeList(LinkList A, LinkList B){
   LinkList C = new LinkList();
   int m =    A.getSize()    ;
   int n =    (1)    ;
   if(  (2)  ){
      int i=1;
      for(;i<n+1;i++){
         LinkNode node1 =  (3)  ;
         LinkNode node2 = B.getElem(i);
        C.append(  (4)  );
        C.append(  (5)  );
      }
      for(;i<m+1;i++){
         LinkNode node1 = A.getElem(i)
        C.append(node1);
      }
   }else{
      int i=1;
      for(;i<m+1;i++){
         LinkNode node1 = A.getElem(i);
         LinkNode node2 = B.getElem(i);
         C.append(node1);
         C.append(node2);
      }
      for(;i<n+1;i++){
         LinkNode node1 =  (6)  ;
          (7)  ;
       }
   }
   (8)  ;
}
```

2. Given a class of binary search tree node as follows:

```
public class BinaryNode {
   public int data;   // data stored in this node
   public BinaryNode leftChild;
   public BinaryNode rightChild;
   public BinaryNode(int data) {
      this.data = data;
      this.leftChild = null;
      this.rightChild = null;
    }
}
```

Define a class for binary search tree, and write a function that prints out the node values for a BST in sorted order **from highest to lowest**.