Student Name:

Student ID:

答题内容写在边线外视为无效

Major:

哈尔滨工业大学深圳研究生院

2008 年 秋 季学期期末考试试卷

HIT Shenzhen Graduate School Examination Paper

Course Name:________________________________Lecturer:___________________

Question	**One**	**Two**	**Three**	**Four**	**Five**	**Six**	**Seven**	**Eight**	**Nine**	**Ten**	**Total**
Mark											

General directions:

This exam is **closed book**. You may not use the text book, your notes, computer, or any other materials during the exam.

No credit will be given for questions left unanswered, so you should be sure to answer all questions, even if you are only taking your best guess.

Write your answer to each question or problem in the paper provided. If necessary, extra sheets will be provided. Make sure your name is written on all of these pages.

Please be sure to write neatly and answer all questions unambiguously.

This exam has a total of _100_ points, and you have 120 minutes.

Time: 09:00-11:00, Monday, Dec. 8, 2008

错误！未找到引用源。 **Single choice [10 points]**

1、Which of the following sorting algorithms is not stable? (B)

(A) Insertion sort (B) Quick sort
(C) Merge sort (D) Bubble sort

2、We say that $f(n)$ is asymptotically larger than $g(n)$ if (D).

(A) $f(n)=O(g(n))$ (B) $f(n)=\Omega(g(n))$

(C) $f(n)=o(g(n))$ (D) $f(n)=\omega(g(n))$

3、An order-statistic tree is an augmented red-black tree. In addition to its usual fields, each node x has a field size[x], which is the number of nodes in the subtree rooted at x , For an order-statistic tree with *n* nodes, the time for insertion, deletion and maintenance of the *size* field are (A)

(A) $O(\lg n)$ $O(\lg n)$ $O(\lg n)$

(B) $O(\lg n)$ $O(\lg n)$ $O(n\lg n)$

(C) $O(\lg n)$ $O(\lg n)$ $O(1)$

(D) $O(\lg n)$ $O(n\lg n)$ $O(n)$

4、There's a B-tree whose minimum degree is t, every node other than the root must have at least __ keys, at most __ keys, every internal node other than the root has at least __ children (D).

(A) t-1 2t t (B) t-1 2t-1 t
(C) t 2t t+1 (D) t-1 2t+1 t

5、Which of the following statements about P, NP,NPC is correct? (C)

(A) P = NP , NPC $\supseteq$ NP (B) P $\supseteq$NP , NPC $\supseteq$ P
(C) P $\subseteq$NP , NPC $\subseteq$NP (D) P = NPC , P $\supseteq$ NP

错误！未找到引用源。 **short answer [30 points]**

6、Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 88, 59 into a hash table of length m = 11 using open addressing with the primary hash function h1(k) = k mod m. Illustrate the result of inserting these keys using linear probing, using quadratic probing with c1 = 1 and c2 = 3. The final results are required to be expressed as charts. [6 points]

0	1	2	3	4	5	6	7	8	9	10
22	88			4	15	28	17	59	31	10

7、Use the arithmetic expression ((a+b)+c*(d+e)+f)*(g+h) to construct a binary tree. [6 points]

8、Explain how to implement two stacks in one array *A*[1 .. *n*] in such a way that neither stack overflows unless the total number of elements in both stacks together is *n*. The PUSH and POP operations should run in *O*(1) time. [6 points]

9、Tell the difference between dynamic programming and greedy programming. [6 points]

10、We are given a directed graph G=(V, E) on which each edge (u, v)∈E has an associated value $r(u, v)$, which is a real number in the range $0 \leqslant r(u, v) \leqslant 1$ that represents the failure rate of a communication channel from vertex u to vertex v. We assume that these probabilities are independent. Give an efficient algorithm to find the most reliable path between two given vertices. [6 points]

错误！未找到引用源。 **[60 points]**

11、Consider the ***searching problem***:

- **Input:** A sequence of n numbers A = 〈a_1, a_2, . . . , a_n〉 and a value v.
- **Output:** An index i such that $v = A[i]$ or the special value NIL if v does not appear in A.

Write pseudocode for ***linear search***, which scans through the sequence, looking for v. Using a loop invariant, prove that your algorithm is correct. Make sure that your loop invariant fulfills the three necessary properties. [10 points]

12、Use a recursion tree to give an asymptotically tight solution to the recurrence $T(n) = T(\alpha n) + T((1 - \alpha)n) + cn$, where α is a constant in the range $0 < \alpha < 1$ and $c > 0$ is also a constant. [12 points]

13、A ***red-black tree*** is a binary search tree with one extra bit of storage per node: its ***color***, which can be either RED or BLACK, and the red-black is a nearly balanced tree. [13 points]

1) A binary search tree is a red-black tree if it satisfies the following ***red-black properties***:

1. Every node is either red or black.
2. The root is black.
3. Every leaf (NIL) is black.
4. ________________(1)________________
5. ________________(2)________________

Please give the missing property 4 and property 5. [2 points]

2) What is the largest possible number of internal nodes in a red-black tree with black-height k? What is the smallest possible number? [3 points]

3) There is the RBT insertion pseudo-code ,which insert a given node ***z*** into a RBT ***T***. Suppose you can use LEFT-ROTATE(***T***, ***z***) and RIGHT-ROTATE(***T***, ***z***) operation directly ,Please fill the blank! [8 points]

```
RB-INSERT($T$, $z$)
  $y \leftarrow nil[T]$
  $x \leftarrow root[T]$
  while $x \neq nil[T]$
      do $y \leftarrow x$
          if key[z] < key[x]
              then x ← left[x]
```

```
            else  ____(1)____
p[z] ← y
if y = nil[T]
    then root[T] ← z
    else if key[z] < key[y]
            then  ____(2)____
            else right[y] ← z
left[z] ← nil[T]
right[z] ← nil[T]
____(3)____
RB-INSERT-FIXUP(T, z)
```

```
RB-INSERT-FIXUP(T, z)
while color[p[z]] = RED
    do if p[z] = left[p[p[z]]]
          then y ← right[p[p[z]]]
               if color[y] = RED
                  then  ____(4)____                 ▹ Case 1
                        ____(5)____                 ▹ Case 1
                        ____(6)____                 ▹ Case 1
                        z ← p[p[z]]                 ▹ Case 1
                  else if z = right[p[z]]
                          then  z ← p[z]            ▹ Case 2
                                ____(7)____         ▹ Case 2
                        color[p[z]] ← BLACK         ▹ Case 3
                        ____(8)____                 ▹ Case 3
                        RIGHT-ROTATE(T, p[p[z]])    ▹ Case 3
          else (same as then clause with "right" and "left" exchanged)
color[root[T]] ← BLACK
```

14、Find an optimal parenthesization of a matrix-chain product whose sequence of dimensions is <5, 10, 3, 12, 5, 50, 6 >. [10 points]

15、 A full k-tree with the depth L has the following properties: All the nodes on the Lth layer are leaves; Every node on other layers has k non-empty subtrees. If we number all the nodes from the first layer to the Lth layer and from left to right with 1,2,3..., please answer the following questions: [15 points]

1) How many nodes does the ith layer have? [3 points]

2) What is the number of the parent of node n, if it exists. [4 points]

3) What is the number of the ith child from left of node n, if it exists. [4 points]

4) In what condition does node n have a right brother? If it has, what is the number of its right brother? [4 points]