編號： 248

國立成功大學 109 學年度碩士班招生考試試題

系　　所：資訊管理研究所

考試科目：資料結構

考試日期：0211，節次：3

※ 考生請注意：本試題不可使用計算機。　請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

1. In each of the following question, please specify if the statement is true or false.
   a. [2%] $n^{1.5} = O(n\ log\ n)$
   b. [2%] A complete binary tree with a height of h can have more nodes than a full binary tree with a height of h.
   c. [2%] When we use a max heap to implement a priority queue, the time complexity of both the add and delete operations are $O(n)$.
   d. [4%] $T(n) = T(n-1) + n,\ T(1) = 1$. Then $T(n) = O(n^3)$

2. In computer science, a priority queue is an abstract data type which is like a regular queue, but each element has a "priority" associated with it. Elements with higher priorities are served before elements with lower priorities.
   a. [5%] Design a data structure in C++ or Java to represent priority queues where both the data and the priority are integers. Note: 1. Only the class definition is required; 2. The size of the queue should only be constrained by the memory space; 3. Your design should allow efficient enqueue and dequeue operations; 4. The use of STL classes is prohibited.
   b. [10%] Implement the enqueue and dequeue operations/methods.

3. Answer the following question regarding Red-Black tree
   a. [2%] Define red-black tree
   b. [8%] What is the big O performance (in terms of the number of nodes in the tree) for the operations find, insert, and remove for a red-black tree in the best and worst cases?
   c. [10%] Show the tree that results from inserting the values 2, 1, 4, 5, 9, 3, 6, 7 into an initial empty red-black tree. Show the tree after each insertion.

4. [5%] Given an expression, such as "$\{[2 \times (a+b)] - c\} \times 3(d+e)$", implement a function in C++ or Java that checks the proper opening and closing of parenthesis.

國立成功大學 109 學年度碩士班招生考試試題

系　　所：資訊管理研究所

考試科目：資料結構

考試日期：0211，節次：3

※ 考生請注意：本試題不可使用計算機。　請於答案卷(卡)作答，於本試題紙上作答者，不予計分。

5、[32%] **True** or **False**, and **EXPLAIN**

Choose **T(true)** or **F(false)**. If the statement is correct, ***briefly state why***. If the statement is wrong, ***explain why or give a counterexample***. Answers **WITHOUT** reasons will get **at most 1 point**.

(a) [4%] (**T, F**) If we want to randomize the ordering of n integers A[1],…,A[n], we can generate another n random numbers B[1],…,B[n] with values in [$-n^5$, n^5], and then sort B[•] in ascending order. We can then use this ordering to reorder A[1],…,A[n]. Therefore, to randomize n integers takes $\Omega(n \log n)$ time.

(b) [4%] (**T, F**) Let $f(x)=\sum_{i=0}^{n} A[i]x^i$, $g(x)=\sum_{j=0}^{m} B[j]x^j$, $f(x)+g(x)=\sum_{k=0}^{\max\{m,n\}} C[k]x^k$, and $f(x)\bullet g(x)=\sum_{l=0}^{mn} D[l]x^l$. Calculating C[] and D[], takes $O(\max\{m,n\})$ and $\Theta(mn)$ time, respectively.

(c) [4%] (**T, F**) Based on (b), if A[] and B[] store their p and q non-zeros in two ordered doubly linked lists in ascending order. Then, to calculate C[] and D[] takes $O(p+q)$ and $\Theta(pq)$ time, respectively.

(d) [4%] (**T, F**) For a binary search tree of n integers, if we insert an integer of new value to the tree, and then delete that integer right away. It is possible the tree structure becomes different from its original form.

(e) [4%] (**T, F**) Given a complete undirected graph K_n with n nodes, let $c_{ij}>0$ represent the length of any edge (i,j). We can use Dijkstra's algorithm to find a shortest simple path that has to pass all n nodes.

(f) [4%] (**T, F**) Given an undirected graph $G=(N,A)$ of $|N|=n$ nodes and $|A|=m$ arcs, let $c_{ij} \geq 0$ represent the length of $(i,j) \in A$. To detect whether there is a zero-length cycle passing node i takes $O(m)$ time.

(g) [4%] (**T, F**) Given a Taipei city street map composed by street segments with n nodes and m arcs, where an arc is a street segment with two adjacent nodes. The average degree for a node is $\Omega(n)$.

(h) [4%] (**T, F**) To build a min-heap of n values, it takes $\Omega(n \log n)$ time.

6、[18%] Given a simple directed network $G=(N,A)$ of $n=|N|$ nodes $A|=m$ arcs, let $c_{ij} \geq 0$, x_{ij} and u_{ij} represent the length, flow, and capacity for $(i,j) \in A$. If n is even and $U=\max_{(i,j)\in A}\{u_{ij}\}$, we want to send a flow of F_k units for $n/2$ pairs $(o_k,d_k)=(k,k+n/2)$, $k=1,...,n/2$. Answer the following questions:

(a) [6%] Briefly explain how to determine whether F_1 can be successfully sent without violating the arc capacity from the origin node 1 to the destination node 1+n/2. What is the complexity of your method?

(b) [6%] Briefly explain how to construct a sub-network $G_k=(N_k,A_k)$ which only contains required nodes and arcs on all possible shortest paths (i.e., the minimum number of arcs) connecting from an origin node o_k to a destination node d_k. What is the complexity of your method?

(c) [6%] Suppose the complexity of your method in (a) is $S(n,m,U)$ time. To send all the flows of F_k units for each $k=1,...,n/2$ at the same time, briefly explain how to determine whether this is possible or not (if we cannot send F_k units for some k). Is there a method that can do this in $O(S(n,m,U))$ time?