國立成功大學

110學年度碩士班招生考試試題

編　號：190

系　所：電腦與通信工程研究所

科　目：資料結構

日　期：0203

節　次：第 2 節

備　註：不可使用計算機

編號： 190

國立成功大學 110 學年度碩士班招生考試試題

系　　所：電腦與通信工程研究所

考試科目：資料結構

考試日期：0203，節次：2

※ 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. (30 **pts**) True or False. Identify all errors in the sentence if it is not correct. You can get the points only when you correct all errors.

   a. Floyd-Warshall algorithm applies the dynamic programming to find all pair shortest paths and can be applied to a graph with negative weights and negative cycles.
   b. There are no back edges, no forward edges, and no cross edges after we apply the breadth-first search algorithm to an undirected graph.
   c. A biconnected graph has only one biconnected component and we can use breath-first search algorithm to find biconnected components in a graph.
   d. The access time of a hash table with open addressing is related to the load factor.
   e. There is no need for parentheses for the prefix notation because there is no ambiguity.
   f. There totally exists n-1 threads in a post-thread binary tree with n nodes, where the left thread of a node points to its predecessor and the right thread of the node points to its successor according to the in-order.
   g. DijKstra's algorithm is a greedy algorithm which can be used to find a single source shortest path. It allows a graph G(V, E) with negative weights. The time complexity is O(E log V) when we use an adjacency matrix to implement it.
   h. The failure function of the Knuth-Morris-Pratt algorithm applied to the pattern "abcabcacab" is shown as follows:

a	b	c	a	b	c	a	c	a	b
0	-1	-1	0	1	2	0	-1	0	1

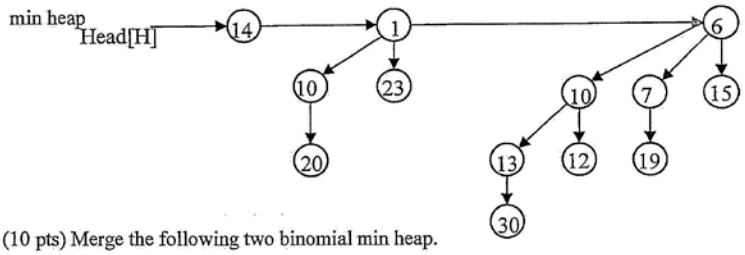
   i. A circular queue can be used to resolve the problem of necessary to move data if a queue is full, where the queue is implemented by an array with size N. But it can only store N-1 data in order to distinguish whether the queue is full or empty. We can pop a data from the location "front" and push a data to the location "rear+1", where "front" and "rear" are pointers to the queue.
   j. If $f(n)=\sum_{i=1}^{n} \log i$, then $f(n)= \theta(\log n)$.

2. (20 **pts**) Heap

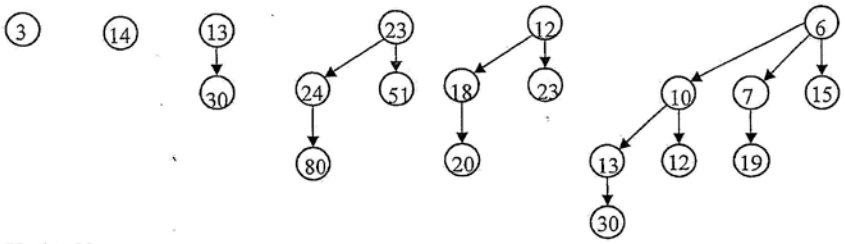
a. (2 pts) Describe the properties of min heap (i.e., two conditions).

b. (4 pts) Describe the properties of a binomial min heap (i.e., four conditions).

c. (4 pts) Please show the resulting figure after you delete the node 1 in the following binomial min heap



d. (10 pts) Merge the following two binomial min heap.



3. (22 **pts**) Hash table

a. (2 **pts**) What is the uniform hashing function?

b. (2 **pts**) What is the perfect hashing function?

Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 88, 59 into a hash table with 11 buckets (i.e., $m = 11$) and each bucket has 1 slot.

c. (2 **pts**) What is the loading factor?

Assume the hash function $h(k) = k \bmod m$. Show the result in the hash table after inserting the above keys with the following:

d. (4 **pts**) Open addressing with linear probing

e. (6 **pts**) Open addressing with quadratic probing, where the rehashing function $rh(k, i) = (h(k) + 2i^2) \bmod m$

f. (6 **pts**) Open addressing with a second hash function $h_2(k) = (m\text{-}1) - (k \bmod (m\text{-}1))$.

4. (10 pts) Selection tree for minimal values

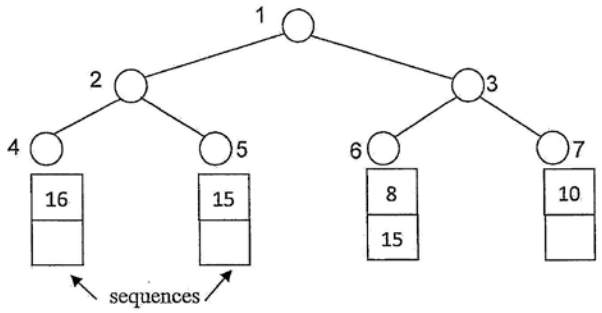
a. (2 **pts**) Show the initial status of the binary tree after the first node in each sequence is processed by a **winner tree.**

b. (2 **pts**) Show the status of the binary tree after the next node is processed by a **winner tree**.

c. (2 **pts**) Show the initial status of the binary tree after the first node in each sequence is processed in a **loser tree.**

d. (2 **pts**) Show the status of the binary tree after the next node is processed by a loser **tree.**

e. (2 **pts**) Compared the two trees, which method is faster? Briefly justify your answer.



5. (18 pts)

a. (10 **pts**) Prove the lower bound of the time complexity of the comparison based sorting algorithm for sorting n elements.

b. (2 **pts**) Let n denote the number of elements. Let m denote the maximal number of digits for all elements, where the each digit ranges from 0 to k.
What is the time complexity of LSD radix sort algorithm?

c. (4 **pts**) Show the result to sort 13 , 5, 10, 9, 6 in the non-decreasing order by the LSD radix sort algorithm, where each number is represented by 4 bits binary number. You have to show the results step by step.

d. (2 **pts**) Is there exist any conflict on the result in a. and b.? Briefly justify your answer.