國立陽明交通大學 110 學年度碩士班考試入學試題

科目：資料結構與演算法(1101)　　考試日期：110 年 2 月 3 日 第 1 節

系所班別：資訊聯招　　第 1 頁，共 8 頁

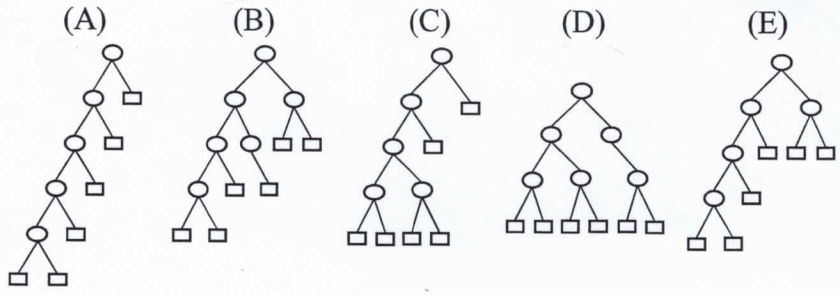
【不可使用計算機】＊作答前請先核對試題、答案卷(試卷)與准考證之所組別與考科是否相符！！

本試卷共有 30 題多選題，每題有一個（含）以上的正確選項。各題填答的選項必須完全符合正確選項，答錯沒有倒扣，若有任一個選項不符合則該題得分為零。各題的分數為：第 1~8 題每題 4 分，第 9~18 題每題 3.5 分，第 19~24 題每題 3 分，第 25~30 題每題 2.5 分。請用答案卡作答。

1. Suppose that problem L_1 is polynomial-time reducible to problem L_2. Which of the following statement(s) is(are) correct?
   (A) If L_1 is NP-hard, then L_2 is NP-hard.
   (B) If L_2 is NP-hard, then L_1 is NP-hard.
   (C) If L_1 is in P, then L_2 is in P.
   (D) If L_2 is in P, then L_1 is in P.
   (E) If a problem is in NP, then there does not exist any polynomial time algorithm to solve it.

2. Assume that $NP \neq P$. Which of the following problem(s) is(are) NP-hard?
   (A) Finding a longest simple path between two nodes in a directed graph where the edge-weight is one for every edge.
   (B) Finding a smallest cycle in a graph where the edge weight is one for every edge.
   (C) Finding a largest cycle in a graph, where the edge weight is one for every edge.
   (D) Finding a negative-weight cycle with any kind (positive or negative) of edge weights.
   (E) Finding a shortest simple path between two nodes in a directed graph with negative-weight cycles.

3. Huffman codes: Given n characters each with a given frequency, find Huffman binary codes for these n characters. Suppose that there are six characters to be encoded. For the following five binary coding trees, select the ones that **cannot** possibly represent Huffman binary codes.



4. Which of the following statement(s) is(are) true about red-black trees?
   (A) Red-black trees are not search trees.
   (B) An external node must be black.
   (C) It is possible that the path from the root to an external node has two consecutive black nodes.
   (D) Every path from the root to an external node has the same number of red nodes.
   (E) It takes $O(1)$ time to insert a key to the red-black tree.

5. Assume that we have a data structure S that contains n elements with $n > 0$. Which of the following statement(s) is(are) true?
   (A) Let S be a linked list with its elements required to be ordered in ascending order. It takes $O(1)$ time to insert a new element into S.
   (B) Let S be a queue. It takes $O(n)$ time to remove all the elements one by one from S.
   (C) Let S be a double-ended queue which is implemented as a doubly linked list and supports insertion and deletion from both ends. It takes $O(1)$ time to add an element to the back of S and $O(1)$ time to find an element of S.
   (D) Let S be a doubly linked list. Given a node of S, it takes $O(n)$ time to delete the node from S.

6. We say that $O(f(n)) < O(g(n))$ if there exists a function in $O(g(n))$ that is greater than all the functions in $O(f(n))$. For n being positive integers, select the correct statement(s) in the following:
   (A) $O(n^{1.1}) > O(n\log^2 n)$
   (B) $O(n!) > O\left(\lfloor n/2 \rfloor^{\lfloor n/2 \rfloor}\right)$
   (C) $O(3^{\log_4 n}) > O(n^2)$
   (D) $O(\log\log n) > O(\log n)$

7. Which of the following graph algorithm(s) utilize dynamic programming?
   (A) Kruskal's algorithm
   (B) Dijkstra's algorithm
   (C) Finding longest common sequence in a pair of strings
   (D) The Floyd-Warshall algorithm for computing the all-pairs shortest path of a graph.

8. The preorder tree walk of a certain binary search tree generates the following sequence of node keys: 6, 2, 4, 3, 5, 9, 7, 8. Which of the following statement(s) is(are) correct?
   (A) 5 is an ancestor of 3.
   (B) 7 has no sibling.
   (C) 5 appears after 3 during the postorder tree walk.
   (D) 8 appears after 7 during the postorder tree walk.
   (E) The two subtrees of the root have the same heights.

9. Consider radix sort of a list of positive integers, with each decimal digit as a key. Assume that we proceed in a least-significant-digit-first manner, and we want to sort the list from small to large. Let the initial list of integers be
   312, 256, 19, 713, 44, 608, 210, 909
   (A) After the first pass, only one of the first four numbers in the list is less than 200.
   (B) The last number in the list after the second pass is 256.
   (C) 44 is among the last four numbers of the list after the second pass.
   (D) It is possible to have radix sort run with linear time complexity with respect to the length of the list to be sorted.

10. Which of the following sorting algorithm(s), using typical implementation, apply the divide-and-conquer approach?
(A) Quick Sort (B) Heap Sort (C) Selection Sort (D) Insertion Sort (E) Merge Sort

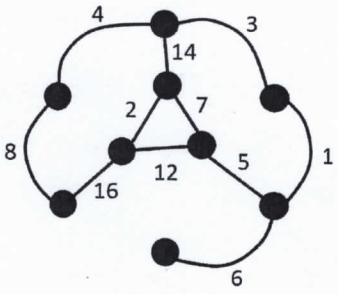
11. Consider the maximum flow problem. Assume that all the edge capacities are positive integers. Which of the following statement(s) is(are) correct?
(A) There is a polynomial time algorithm to find a maximum cut.
(B) There is a polynomial time algorithm to find a minimum cut.
(C) There are possibly many distinct flows having maximum flow value with different edge flow on edges.
(D) There are possibly many distinct minimum cuts with different total capacities.
(E) The sum of the capacity of a minimum cut and the value of a maximum flow must be an even number.

12. Which of the following statement(s) is(are) correct? Assume that $T(1)$ is a constant and $n>0$.
(A) If $T(n)=2T(n/2)+n\log n$, then $T(n)=\Theta(n\log n)$.
(B) If $T(n)=2T(n/2)+n/\log n$, then $T(n)=\Theta(n)$.
(C) If $T(n)=2T(n/2)+n/\log^2 n$, then $T(n)=\Theta(n)$.
(D) If $T(n)=\sqrt{n}\,T(\sqrt{n})+n$, then $T(n)=\Theta(n\log n)$.
(E) If $T(n)=T(\frac{n}{2}+\sqrt{n})+n$, then $T(n)=\Theta(n)$.

13. Let $G=(V,E)$ be an undirected connected graph, where each edge e has a given length $l(e)>0$. Let s be a fixed vertex of this graph and let $\delta(v)$ be the length of a shortest path from s to v. Which of the following statement(s) is(are) correct?
(A) For each vertex $v\neq s$, there is an incident edge $e=(u,v)$ such that $\delta(v)=\delta(u)+l(e)$.
(B) For each vertex $v\neq s$, choose one incident edge $e=(u,v)$ such that $\delta(v)=\delta(u)+l(e)$; the collection of these edges forms a spanning tree of G.
(C) For each vertex $v\neq s$, choose one incident edge $e=(u,v)$ such that $\delta(v)=\delta(u)+l(e)$; the collection of these edges forms a minimum weight spanning tree of G.
(D) $\delta(v)>\delta(u)+l(e)$ for some edge $e=(u,v)$.
(E) $\delta(v)\leq\delta(u)+l(e)$ for every edge $e=(u,v)$.

14. Consider a B-tree with minimum degree of 3. Keys in the same node are stored in ascending order. The median key is used to split a full node. Initially the B-tree is empty. We then build the B-tree by inserting the keys 5, 6, 4, 2, 1, 7, 8, and 10 one by one in the given order. The following statements are independent from each other and all start from the same tree. Which of the following statement(s) is(are) true?
(A) If 9 is inserted into the B-tree, the height of the B-tree is changed.
(B) If 3 is inserted into the B-tree, the height of the B-tree is not changed.
(C) The keys 7 and 10 are not in the same node.
(D) The keys 1 and 2 are not in the same node.

15. For the given weighted graph to the right, let x and y be the total weights of its minimum spanning tree and its second best minimum spanning tree, respectively.



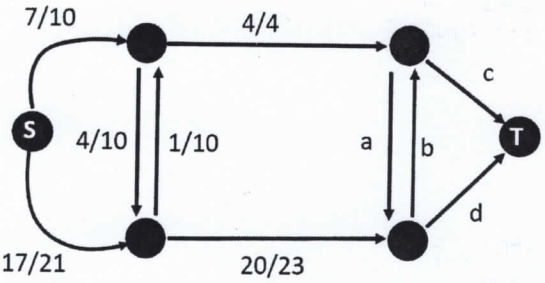
Which of the following statement(s) is(are) true?

(A) $y - x = 7$
(B) $y - x = 5$
(C) $x + y = 77$
(D) $2y - x = 51$
(E) $2x - y = 31$

16. This question is about binary search trees. All the keys are unique. Let x be a tree node, *parent*[x] be the parent of x, *left*[x] be the left child of x, *right*[x] be the right child of x, and *key*[x] be the key of x. Assume that the root of a binary search tree has left and right children. The key of the root is greater than the key of the left child but is smaller than the key of the right child. Which of the following statement(s) is(are) true for binary search trees?

(A) If *left*[x] and *left*[*left*[x]] are non-empty, then *key*[x] is greater than all the keys in the subtree rooted at *left*[x].
(B) If *parent*[x] is non-empty, then we must have *key*[*parent*[x]] > *key*[x].
(C) If *left*[*left*[x]] and *left*[*right*[x]] are non-empty, then *key*[*left*[*left*[x]]] must be larger than *key*[*left*[*right*[x]]].
(D) If *parent*[x], *right*[x], and *right*[*right*[x]] are non-empty, then *key*[*parent*[x]] must be less than all the keys in the subtree rooted at *right*[x].

17. This question is about a directed flow network. The flow and capacity are denoted as x/y on an edge, where x is the flow amount and y is the capacity. The following figure shows a flow in a directed flow network. S and T are the source and the sink, respectively.



Which of the following set(s) of a, b, c, and d is(are) valid for the flow?

(A) a = 1/10, b = 9/10, c = 11/15, and d = 13/17
(B) a = 2/7, b = 9/8, c = 12/12, and d = 12/14
(C) a = 3/4, b = 10/10, c = 11/13, and d = 13/21
(D) a = 5/5, b = 8/10, c = 7/11, and d = 17/18

10/10 TLOU2

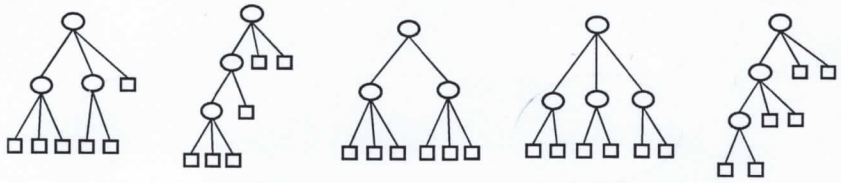
18. Divide and conquer for sorting: Given a set of n integers, where n is exact power of 2:
   (A) If I divide the n integers into 4 equal parts, conquer the 4 parts recursively, and merge the 4 parts into a sorted sequence in linear time, I get the sorted sequence in $O(n \log n)$ time.
   (B) If I divide the n integers into 2 equal parts, sort the 2 parts using selection sort, I get the sorted sequence in $O(n^2)$ time.
   (C) If I divide the n integers into 2 equal parts, sort the 2 parts using heap-sort, I get the sorted sequence in $O(n \log n)$ time.
   (D) Let $\lg n$ represent base-2 logarithm. If I divide the n integers into $\left\lceil \frac{n}{\lg n} \right\rceil$ parts with each part except the last part having m integers ($m \le \lg n$), sort each part using heap-sort, and merge the $\left\lceil \frac{n}{\lg n} \right\rceil$ parts, I get the sorted sequence in $O(n \log \log n)$ time.

19. A maximum binary heap (Max-Heap for short) is implemented in an array, starting with index 1. (In many programming languages, the array index starts at 0. In this case, we skip the first cell which has index 0. For example, [-,3,2,1] represents a Max-Heap where 3 is the root, 2 is the left child of 3, and 1 is the right child of 3. Since we don't use the first cell, it is marked as '-'.) Which of the following statement(s) is(are) correct?
   (A) Given the input A = [-,1,2,3,4,5,6,7,8,9], the Max-Heap built by the best (linear-time) algorithm from A is [-,9,8,6,7,3,2,5,1,4].
   (B) B = [-,9,8,7,4,5,6,3,1,2] is a Max-Heap. By deleting the maximum, we get B = [-,8,5,7,4,2,6,3,1]
   (C) C = [-,8,5,7,4,2,6,3,1] is a Max-Heap. By inserting 10, we get C = [-,10,8,7,4,5,6,3,1,2].
   (D) Binary Max-Heap supports the merging of two Max-Heap in linear time.

20. This question considers "vector" in C++ STL. The usage of vector is very similar to a regular array in C/C++ in that it supports direct access. In addition, it also supports the operation **`push_back`**, which adds an element at the end of a vector. As a result, we do not need to declare the size for a vector.
   (A) Elements in a vector are stored in consecutive memory spaces so that it supports direct access.
   (B) Let **`v`** be a vector of size m. The statement **`v.push_back(a)`** adds **`a`** at the end of **`v`**. However, the memory allocated for **`v`** might be too small to hold the added element **`a`**. This is handled by allocating a new block of memory of size (m+1), copying all the elements of **`v`** to the newly allocated space, and then inserting **`a`** at the end of **`v`**.
   (C) If we make a sequence of n insertions, there is a way that we can achieve $\Theta(n)$ amortized cost for the memory space management.
   (D) A single **`push_back`** operation could take $\Theta(n)$ time, where n is the number of elements in the vector.

21. This question is about generalizing Huffman's algorithm to ternary codes (i.e., codes using three symbols 0, 1, and 2). Suppose that there are six characters to be encoded. For the following five ternary coding trees, select the ones that **cannot** possibly represent Huffman ternary codes.

(A) (B) (C) (D) (E)



22. Let $f(n)$ and $g(n)$ be asymptotically positive functions, where $\log(g(n)) \geq 1$ and $\log(f(n)) \geq 1$ for all sufficiently large n. Which of the following statement(s) is(are) correct?

(A) $f(n) = O(g(n))$ implies $\log(f(n)) = O(\log(g(n)))$

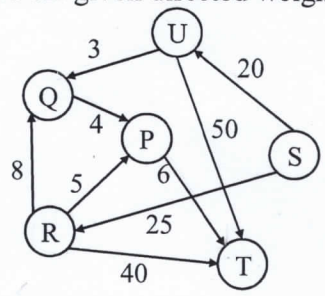
(B) $\log(f(n)) = O(\log(g(n)))$ implies $f(n) = O(g(n))$

(C) $f(n) = O(g(n))$ implies $2^{f(n)} = O(2^{g(n)})$

(D) $\lceil \log n \rceil ! = O(n)$

(E) $2^{\sqrt{2 \log n}} = O(n)$

23. For the given directed weighted graph below:



(A) In every topological sort of the graph, R has to appear after U.

(B) The path from S to T with the most total weight includes four edges.

(C) The path from S to T found by Dijkstra's algorithm contains three edges.

(D) If we add an edge from P to U with weight=0, we can still obtain a valid topological sort.

(E) If we add an edge from P to U with weight=0, the Dijkstra's algorithm will still find the same path from S to T as before the edge is added.

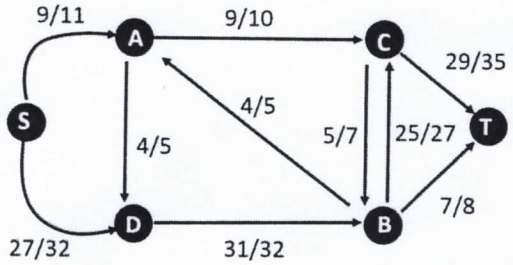
24. The hash function [$h(key) = key \bmod m$] is used to map any integer key to one of the slots, indexed 0, …, $m-1$, of a hash table of m slots.
(A) If m is 5, there is just one occurrence of collision among these keys: 1, 2, 4, 6, 8.
(B) Collision does not happen as long as the load factor is less than 0.5.
(C) Consider a hash table where m is 5 and uses chaining to handle collisions. The table is initially empty. After we insert the keys 1, 2, 4, 6, 8 into the table in the given order, key 8 is in the slot indexed 3.
(D) Consider a hash table where m is 5 and uses open dressing with linear probing to handle collisions. The table is initially empty. After we insert the keys 1, 2, 4, 6, 8 into the table in the given order, key 8 is in the slot indexed 0.

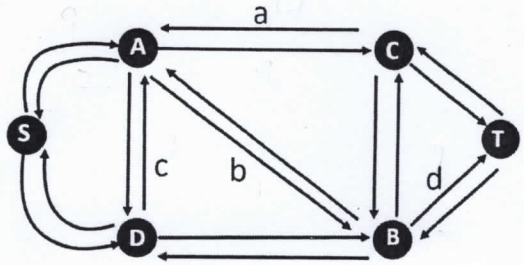
25. A red-black tree is initially empty. Now four keys are inserted in the order of 5, 3, 8, 2. Which of the following statement(s) is(are) true about the resulting red-black tree?
(A) There are no more than 4 external nodes.
(B) There are two red nodes.
(C) The node 8 is black.
(D) Exactly one rotation operation occurs during the insertion of the four nodes.
(E) Nodes 2 and 3 have different colors.

26. Consider the following sequence of operations for disjoint sets (there is no set initially):
Make-Set('a'), Make-Set('b'), Make-Set('c'), Union('b','c'), Make-Set('d'), Make-Set('e'), Make-Set('f'), Union('e','f'), Union('a','b'), Union('a','f')
Which of the following statement(s) is(are) true after this sequence of operations?
(A) There are two sets.
(B) Find-Set('b') and Find-Set('f') return the same node.
(C) Find-Set('c') and Find-Set('d') return the same node.
(D) Each union operation takes constant time.

27. Activity-selection problem. Suppose we have a set $S = \{a_1, a_2, ..., a_n\}$ of n proposed activities. Each activity a_i has a start time s_i and a finish time f_i, where $0 \le s_i < f_i < \infty$. If selected, activity a_i takes place during the half-open time interval $[s_i, f_i)$. Activities a_i and a_j are compatible if the intervals $[s_i, f_i)$ and $[s_j, f_j)$ do not overlap. That is, a_i and a_j are compatible if $s_i \ge f_j$ or $s_j \ge f_i$. In the activity-selection problem, we wish to select a maximum-size subset of mutually compatible activities. For the following greedy ideas, select the one(s) that yield an optimal solution.
(A) Selecting the first activity to start (that is compatible with all previously selected activities).
(B) Selecting the first activity to finish (that is compatible with all previously selected activities).
(C) Selecting the last activity to start (that is compatible with all previously selected activities).
(D) Selecting the last activity to finish (that is compatible with all previously selected activities).
(E) Selecting the compatible activity that overlaps the fewest other remaining activities.

28. This question is about the Ford-Fulkerson method. The flow and capacity is denoted as x/y on an edge, where x is the amount of flow and y is the capacity. A flow network is shown below to the left, and its corresponding residual network is show below to the right:





Which of the following statement(s) is(are) true?

(A) a – b = 5, c – d = 2

(B) b = c, a – d = 8

(C) a + b = c + 10d

(D) a + b + c = 17, a + c = 13

29. Function iteration: The notation $f^{(i)}(n)$ (NOT $f^{i}(n)$) denotes the function $f(n)$ iteratively applied i times to an initial value of n. For non-negative integers i, $f^{(i)}(n)=n$ if $i=0$, and $f^{(i)}(n)=f(f^{(i-1)}(n))$ if $i>0$. Now let $\lg n$ represent base-2 logarithm. $\lg^{(i)} n$ is defined as above with $f(n)=\lg n$; note that $\lg^{(i)} n$ is defined only if $\lg^{(i-1)} n>0$. Let $\lg^{*} n$ denote the iterated logarithm, given by $\lg^{*} n=\min\{i\geq 0:\lg^{(i)} n\leq 1\}$. Which of the following statement(s) is(are) correct?

(A) If $f(n)=2n$, then $f^{(i)}(n)=2^{i}n$.

(B) If $f(n)=2n$, then $f^{(i)}(n)=2n^{i}$.

(C) $\lg^{*} n=\lg\lg n$

(D) $\lg^{*} 16=3$

30. Given n integers, the problem of searching for the k'th largest integer among the n integers is called "selection problem". Selection problems can be solved in linear time. Typical implementation starts with partitioning the n integers into $\lceil n/5\rceil$ groups. Then we solve a sequence of median finding (selection problem again) from integers of smaller problem size. Finally we can drop out more than a quarter of the n integers. The next iteration is to deal with a selection problem of size less than (3/4)n. Thus linear time is achieved. Which of the following statement(s) is(are) correct?

(A) In the linear time algorithm stated above, if we partition the set of n integers into $\lceil n/3\rceil$ groups, we still can achieve linear time performance.

(B) In the linear time algorithm stated above, if we partition the set of n integers into $\lceil n/7\rceil$ groups, we still can achieve linear time performance.

(C) This problem can be solved in $O(n\log n)$ time using heap sort.

(D) If a heap sort is applied, this problem has lower bound $\Omega(n\log n)$ because $\Omega(n\log n)$ is the lower bound to heap sort.