※ 注意：請用 2B 鉛筆作答於答案卡，並先詳閱答案卡上之「畫記說明」。

(一) 單選題 (共五小題，每題答對得 8 分，答錯倒扣 2 分，倒扣至單選題總分為 0 分為止)

1. Which of the following descriptions about graph is NOT correct?

(A) A planar graph has at most two vertices that have odd number of adjacent edges.

(B) A bipartite graph is always two-colorable.

(C) A non-empty graph must contain a non-empty clique.

(D) A complete graph with n vertices has n(n - 1)/2 edges. (Note: n > 0)

(E) An edge in a hyper graph may be incident to any number of vertices.

2. Consider a special singly linked list in which the data in the list are always sorted. Suppose the access time to the first (called "head") and last (called "tail") elements of the list is O(1) (e.g. by C/C++ pointers to "head" and "tail") and must be kept O(1) after any list operation. Let the smallest and largest elements be stored in "head" and "tail", respectively. Which of the following descriptions about this linked list is NOT correct?

(A) We cannot perform binary search with O(log n) runtime to find a data in the linked list even though all the data are sorted.

(B) It takes O(1) time to find the second smallest element in the list.

(C) It take O(1) time to check if the linked list is empty.

(D) It takes O(n) time to insert a data to the linked list.

(E) It takes O(1) time to delete the largest element from the linked list.

3. Let *goo*(*m*) be a function with runtime complexity O(*m*) for the parameter *m*. What is the runtime complexity of the following function *foo*(*n*)?

```
void foo(int n) {
   for (int i = 0; i < n; i++) {
      for (int j = i; j >= 0; j = j/2)
         goo(j);
   }
}
```

見背面

(A) O(n)

(B) O(n log n)

(C) $O(n^2)$

(D) $O(n^2 \log n)$

(E) $O(n^3)$

4. Suppose we use an AVL tree to store the characters in an English word. The characters are inserted in the order as they are presented in the word and compared by their corresponding ASCII code (e.g. 'A': 65, 'B': 66, 'a': 97). If repeated character is to be inserted, we just skip it. Which of the following words will result in the most number of rotations? (Note: we count one rotation for a single rotation, and two rotations for a double rotation.)

(A) Hello

(B) World

(C) modern

(D) national

(E) taiwan

5. A relation "$\preccurlyeq$" is a "partial order" on a set S if it has the following properties:
   - Reflexivity: $a \preccurlyeq a$ for all $a \in S$
   - Antisymmetry: $a \preccurlyeq b$ and $b \preccurlyeq a$ implies $a = b$.
   - Transitivity: $a \preccurlyeq b$ and $b \preccurlyeq c$ implies $a \preccurlyeq c$

   Which of the following is the best data structure to store the elements in a partial order set S?

(A) An array A where $A[\,i\,] \preccurlyeq A[\,j\,]$ if index $i \leq j$.

(B) A binary search tree B in which for an arbitrary node n in the tree, $l \preccurlyeq n$ if l is the left child of the node n, and $n \preccurlyeq r$ if r is the right child of n.

(C) An queue Q where $Q[\,i\,] \preccurlyeq Q[\,j\,]$ if index $i \leq j$.

(D) A directed acyclic graph D in which for an arbitrary node n in the graph, $c \preccurlyeq n$ if c is an child node of n.

(E) A hash H with hash function f where elements $m \preccurlyeq n$ if $f(m) \leq f(n)$.

接次頁

(二) 複選題 (共六小題，每題 10 分, 題內每個選項單獨計分，答對得 2 分，答錯倒扣 2 分，倒扣至複選題總分為 0 分為止。如某題未作答則該題得 0 分，不倒扣。)
(例：正確答案為 ABC，若答 BCD，則答對三個選項(BCE)，答錯兩個選項(AD)，故共計得 2 分)

6. Which of the following is(are) correct?

(A) Given two programs with time complexities $O(n^d)$ and $O(c^n)$, where d and c are both greater than 1. For the same input test data, the first program must run faster (i.e. complete earlier) than the second one for large enough n.

(B) Given a program with exponential runtime complexity $O(2^n)$ and linear amount of memory usage O(n), we can always rewrite it into an equivalent program that requires exponential amount of memory usage but run with linear time efficiency. (Note: Two programs are called "equivalent" if and only if they always respond the same for the same input data).

(C) If the function "void foo(int m)" has quadratic runtime complexity $O(m^2)$, then the following C program:

"int main() { for (int i = 1; i < n; i++) foo(i * i); return 1; }"

has cubic time complexity $O(n^3)$.

(D) Let p(n) be a linear function, then $p(n) = \Omega(\log^2 n)$.

(E) Let f(n) and g(n) be two functions of n. If "f(n) = O(g(n))" and "g(n) = Ω(f(n))", then f(n) = Θ(g(n)).

7. For the following storages and operations on the students' school records, which is(are) correct?

(A) Suppose we use a BIG array (with the size equal to the number of students) to store all the students' records. We then sort the records based on the students' ID numbers (an 8-digit number) so that we can access any record in O(log(n)) time later (e.g. by binary search). In this case, the "quick sort" algorithm is not a good choice because it is not a stable sorting algorithm.

(B) Suppose we use a doubly linked list to store the students' records, and the records are stored in the list with the ascending order of the students' ID numbers. If one student is going to drop out of school, it takes O(n) time to remove his/her record from the list (n = number of students).

見背面

(C) Suppose we use a max-heap to store the students' records, with the comparison function "GET(u, v)" on the student's accumulated hours of courses taken. That is, GET(u, v) returns true if student u has taken more hours of courses than v. Suppose the president announces a weird rule that the "president award" goes to the student who has taken the maximum number of hours of courses, then it takes O(1) time to identify such a student.

(D) Suppose we use a directed graph to store the friendship between students: each vertex represents a student, and each edge <u, v> (i.e. from u to v) implies that the student u is acknowledged as a friend of v (i.e. v grants the friendship to u). Assuming that every student has at least granted one friendship to another student, and has been acknowledged as a friend by yet another student, then the graph of the students form a "strongly connected component (SCC)".

(E) Suppose we use an undirected graph to store the friendship between students: each vertex represents a student, and each edge (u, v) records that u and v are mutual friends. If we impose the following transitivity rule: "if u and v are mutual friends, and v and w are mutual friends, then u and w must also be mutual friends", then the graph is a union of disjoint cliques.

8. Which of the following is(are) true about the "tree" data structure?

(A) Let T_a and T_b be two trees with roots r_a and r_b. Let n_a and n_b be the numbers of nodes for T_a and T_b, respectively. Let's compose a new tree T_c by creating a new node r_c with two children r_a and r_b. Then T_c must have exactly $(n_a + n_b)$ number of edges.

(B) For an n-ary tree (i.e. each internal node has exactly n child nodes) that has v number of internal nodes (v != 0), the number of leaf nodes (i.e. the external nodes) is equal to "v(n-1) + 1".

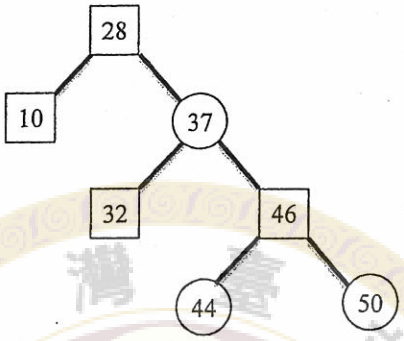
(C) The height of a binary tree is $O(\log_2 n)$, where n is the number of nodes.

(D) If the height of the tree is smaller than n/3, where n is the number of nodes, the tree must be a ternary tree.

(E) The number of paths from the root node to the leaf nodes is proportional to the number of nodes in the tree.

接次頁

9. Given the red-black tree below. Let the square node be a black node, and the circle node be red. Which of the following is(are) correct for the specified operation(s)?



(A) After inserting a new node "30", the root node is still "28".

(B) After inserting a new node "40", the node "32" will become red.

(C) After deleting the node "50", the node "37" will become root.

(D) After inserting a new node "60", the node "37" will become root.

(E) After deleting the node "10", the node "46" will now become red.

10. Which of the following is(are) correct?

(A) Figure 10-1 has a 4-clique.

(B) Figure 10-1 is not a graph.

(C) Figure 10-2 is the complement graph of Figure 10-3.

(D) Figure 10-2 and Figure 10-3 are isomorphic.

(E) Figure 10-4 is a planar graph.

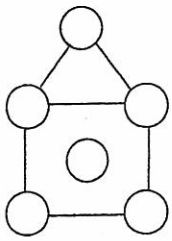


Figure 10-1

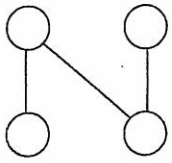


Figure 10-2

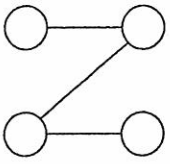


Figure 10-3

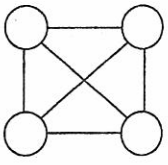
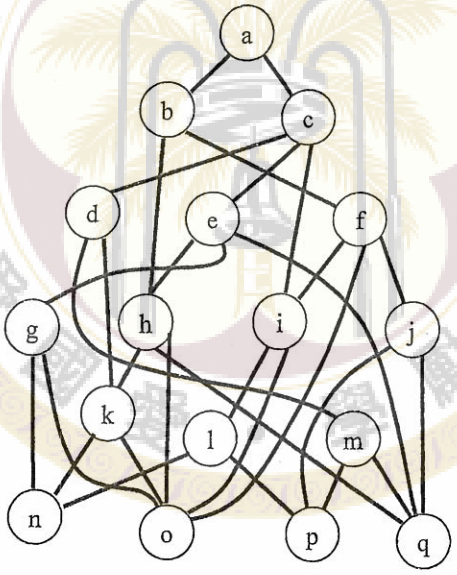


Figure 10-4

見背面

11. For the following graph, each node (except for the leaf nodes) has at least two child nodes and is placed above all of its children on the figure. For a node with n children, we call the child on the leftmost edge the "first child", the child on the second leftmost edge the "second child",..., and the child on the rightmost edge the "n-th child". For example, h and f are b's first and second children, and h, g and q are e's first, second, and third children, respectively. Let's perform "post-order traversal" from the root node a, and number the visited nodes with ascending numbers (i.e. 1, 2, ..., etc). That is, for each node, we first check if its first child has been visited. If yes, we skip it and continue to the second child. If not, we recursively perform the traversal for the children of this child node. We mark the node "visited" and number it if all of its children have been visited. After the traversal, let the function "Num()" return the number of the node. For example, Num(a) = 17. Which of the following is(are) correct?



(A) Num(i) = 8

(B) Num(d) > Num(f)

(C) Num(q) = 12

(D) Num(b) + Num(e) = 25

(E) Num(f) = 9

試題隨卷繳回