※ 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」。

1. For each of the following algorithms in the next five problems, what is the tightest asymptotic upper bound for its runtime complexity for $n$ numbers? Please use the following table of (A)-(E) choices when answering.

| (A) | (B) | (C) | (D) | (E) |
|---|---|---|---|---|
| $O(1)$ | $O(\lg n)$ | $O(n)$ | $O(n \lg n)$ | $O(n^2)$ |

(3 points) Selection sort: expected time?

2. (3 points) Merge sort: expected time?

3. (3 points) MAX_HEAPIFY for a max-heap: expected time?

4. (3 points) Quick sort: worst-case time?

5. (3 points) Bucket sort when there are $\Theta(n)$ buckets: expected time?

6. (5 points) Which of the following is NOT an in-place sorting algorithm (in their standard implementations)?

(A) bubble sort
(B) insertion sort
(C) merge sort
(D) heap sort
(E) all choices are in-place sorting algorithms

7. (5 points) In a *max-heap* stored in an array, after these 6 numbers have been inserted in this exact sequence: 5, 4, 2, 6, 1, 3, what is the value of the node at index 2 (with the root being the node at index 1)?

(A) 1
(B) 2
(C) 4
(D) 5
(E) none of the other choices

8. (5 points) Following the previous problem, what is the value of the left child of the node with value 3?

(A) 1
(B) 2
(C) 4
(D) 5
(E) none of the other choices

9. (5 points) Consider an array of 8 elements being sorted using quicksort. It has just finished the first pass of partitioning and pivot swapping, thus changing the original array into the following array:

$$[7, 11, 16, 10, 17, 1, 18, 30]$$

How many elements could have been the pivot?

(A) 1
(B) 2
(C) 3
(D) 4
(E) none of the other choices

見背面

10. (5 points) Consider an integer array $A$ of size $n > 0$ with elements $A[1], \ldots, A[n]$, where the first $p$ elements are positive and the other $(n-p)$ elements are negative. The following algorithm calculates $p$ correctly by calling COMPUTE-P$(A, n)$. What loop invariant is maintained for the **while** loop of the algorithm?

```
COMPUTE-P(A, len)
if A[1] < 0 return 0
left = 1, right = len
while left < right
    m = ceiling((left+right)/2)
    if A[m] > 0
        left = m
    else
        right = m - 1
return left
```

(A) $left + 1 \leq p \leq right$
(B) $left + 1 \leq p \leq right - 1$
(C) $left \leq p \leq right$
(D) $left \leq p \leq right - 1$
(E) none of the other choices

11. (5 points) Consider a string of length $n$, e.g., cbaabcba. We want to add the smallest number of characters to the front of the string to make it a palindrome, e.g., abcbaabcba. This problem can be solved by first concatenating the string with its inverse with a delimiter #, e.g. cbaabcba#abcbaabc. Then, run the pre-processing routine of the Knuth-Morris-Pratt algorithm on the new string $s_0 s_1 \cdots s_{2n}$ of length $(2n+1)$ to get a failure function $f_s(j)$, which computes the length of the longest prefix of the string $s_0 s_1 \ldots s_j$ that is also a suffix. After running the pre-processing algorithm, how can we calculate the smallest number of characters that should be added to the front of the original string to make a palindrome?

(A) $n - f_s(n-1) + 1$
(B) $n - f_s(n)$
(C) $n - \max_{0 \leq j \leq n-1}(f_s(j))$
(D) $n - \max_{n+1 \leq j \leq 2n}(f_s(j))$
(E) none of the other choices

12. (5 points) Which of the following statement has been proved to be true?

(A) An NP problem of size $n$ must take more than $poly(n)$ time to solve.
(B) Any NP problem is polnomial-reducible to any NP-complete problem.
(C) Any NP-complete problem is polnomial-reducible to any NP problem.
(D) If an NP-complete problem can be solved by an approximation algorithm with ratio 2, any other NP-complete problem can also be solved by an approximation algorithm with a constant ratio.
(E) none of the other choices

13. (5 points) Consider a hash table with $f(n)$ entries with a hash function that can uniformly dispatch items to those entries. Which property below ensures that a successful search can be done in $O(1)$?

(A) $f(n) = O(\sqrt{n})$
(B) $f(n) = O(n)$
(C) $f(n) = \Omega(\sqrt{n})$
(D) $f(n) = \Omega(n)$
(E) none of the other choices

接次頁

14. (5 points) Consider a red-black tree with $n$ internal nodes, where $n$ is even. At most how many of them can be a black node with one red child?

    (A) $\frac{n}{2}$
    (B) $n$
    (C) $\lfloor \log_2(n+1) \rfloor - 1$
    (D) $\lfloor \log_2(n+1) \rfloor$
    (E) none of the other choices

15. (5 points) In the red-black tree with 26 nodes that reaches the solution of the problem above, what is the maximum height of the tree? A tree with one node is assumed to have height 1.

    (A) 5
    (B) 6
    (C) 7
    (D) 8
    (E) 9

16. (5 points) Dynamic programming can be used to solve the matrix-chain multiplication problem. Suppose we hope to compute the matrix product $A_1A_2 \dots A_6$ with the matrix dimensions as follows.

| matrix | $A_1$ | $A_2$ | $A_3$ | $A_4$ | $A_5$ | $A_6$ |
|---|---|---|---|---|---|---|
| dimension | $30 \times 35$ | $35 \times 15$ | $15 \times 5$ | $5 \times 10$ | $10 \times 20$ | $20 \times 25$ |

    Let $m[i,j]$ be the minimum number of scalar multiplications needed to compute the matrix $A_iA_{i+1} \dots A_j$. What is $m[2,5]$?

    (A) 4375
    (B) 7125
    (C) 5375
    (D) 8500
    (E) none of the other choices

17. (5 points) What is the time complexity of finding the minimum number of scalar multiplications needed for $A_1A_2 \dots A_n$ using dynamic programming?

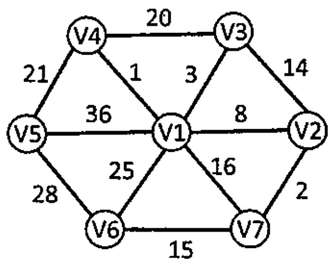
    (A) $O(n^{2.5})$
    (B) $O(n^3)$
    (C) $O(n^2)$
    (D) $O(n)$
    (E) none of the other choices

18. (5 points) When using Dijkstra's algorithm to solve the single-source shortest problem for a graph having $V$ vertices and $E$ edges, which is INCORRECT?

    (A) The search principle is breadth-first.
    (B) It is a greedy algorithm.
    (C) When using an array to implement the min-priority queue, the time complexity is $O(V^2)$.
    (D) For a dense graph where $E = \Theta(V^2)$, the algorithm can run in $O(V^2)$ when using the binary min-heap to implement the min-priority queue.
    (E) none of the other choices

見背面

19. (5 points) Consider the following undirected graph $G$. Let $T$ be the minimum spanning tree (MST) of the graph. Which is INCORRECT?




(A) Edge $(V_1, V_4)$ is in $T$
(B) Edge $(V_4, V_5)$ is in $T$
(C) Let $w_{4,5}$ be the weight of $(V_4, V_5)$. If we decrease $w_{4,5}$ by 16 (i.e., $w_{4,5}$ becomes 5), then $T$ remains to be the MST of $G$
(D) Let $w_{2,3}$ be the weight of $(V_2, V_3)$. If we decrease $w_{2,3}$ by 10 (i.e., $w_{2,3}$ becomes 4), then $T$ remains to be the MST of $G$
(E) none of the other choices

20. (5 points) Given a graph $G = (V, E)$ with the adjacent matrix $W = w_{ij}$ with $W$ an $n \times n$ matrix. The following algorithm can compute the All-Pairs Shortest Path of $G$.

```
FLOYD-WARSHALL'(W)
D = W
for k = 1,2,...,n
    for i = 1,2,...,n
        for j = 1,2,...,n
            d_ij = min(d_ij, d_ik + d_kj)
```

What is the SPACE complexity of this algorithm?

(A) $\Theta(n^2)$
(B) $\Theta(n^3)$
(C) $\Theta(n)$
(D) $\Theta(n^4)$
(E) none of the other choices

21. (5 points) When using the Edmonds-Karp algorithm to find the maximum flow of a flow network $G = (V, E)$, which is CORRECT?

(A) The time complexity is $O(E|f^*|)$, where $f^*$ denotes a maximum flow in the network.
(B) The algorithm uses the depth-first search principle.
(C) The time complexity is $O(VE^2)$.
(D) The time complexity is $O(VE|f^*|)$.
(E) none of the other choices

22. (5 points) Consider a fractional knapsack problem of 6 items. The $i$-th item is worth $v_i$ dollars and weights $w_i$ pounds.

| item $i$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| ponds $w_i$ | 4 | 2 | 8 | 5 | 5 | 8 |
| dollors $v_i$ | 3 | 8 | 16 | 7 | 9 | 20 |

Suppose that at most $W$=20 pounds can be carried in the knapsack. The sequence of picking the items is

(A) 2, 6, 3, 5
(B) 2, 6, 5, 3
(C) 2, 6, 5, 4, 3
(D) 6, 3, 5, 2, 4
(E) none of the other choices

試題隨卷繳回