中原大學 97 學年度碩士班入學考試

4 月 13 日 14:00~15:30 資訊工程學系

誠實是我們珍視的美德，
我們喜愛「拒絕作弊，堅守正直」的你！

科目：資料結構與演算法

□可使用計算機，惟僅限不具可程式及多重記憶者　■不可使用計算機

注意：請勿在本試題卷上作答，按順序於答案卷上作答。

I. 選擇題 (40%) (單選題，每題兩分，請標明題號及答案)

1. Consider the recursive algorithm for computing the function $F(n)$ as follows:

```
int F(int n) {
    if (n <= 1) return(1);
    else        return F(n-1) + n*n;   }
```

What's the returned value of the function call $F(5)$?

(A) 15 (B) 25 (C) 55 (D) 120 (E) None of the above.

2. Consider the function given by: $T(n) = \sum_{k=1}^{n} \frac{1}{k} + \sum_{k=1}^{\lg n} k$, then $T(n)$ =?

(Note: lg is log of base 2).

(A) $\Theta(\lg n)$ (B) $\Theta(\lg\lg n)$ (C) $\Theta(\lg^2 n)$ (D) $\Theta(n)$ (E) None of the above.

3. Consider the recurrence given by: $T(n) = T(\sqrt{n}) + \lg n$, then $T(n)$ = ?

(A) $\Theta(\lg n)$ (B) $\Theta(\lg\lg n)$ (C) $\Theta(\lg^2 n)$ (D) $\Theta(n)$ (E) None of the above.

4. Which of the following statement is **not** correct?

(A) $\lg(n!) = O(n^2)$ (B) $\lg\lg n = O(\lg^2 n)$ (C) $n! = \Omega(2^n)$

(D) $5n^2 - 6n = \Theta(n^2)$ (E) $2^{\lg n} = \Omega(n^2)$

5. Which of the following recurrence is for the worst-case running time of Merge Sort?

(A) $T(n) = 2T(n/2) + \Theta(1)$ (B) $T(n) = 2T(n/2) + \Theta(n)$

(C) $T(n) = T(n-1) + \Theta(1)$ (D) $T(n) = T(n/2) + \Theta(n)$

(E) None of the above.

6. Which of the following sorting algorithm is the most efficient in the worst-case?

(A) Insertion Sort (B) Selection Sort (C) Shell Sort (D) Merge Sort

(E) Quick Sort.

7. A sorting algorithm is said to sort *in place* if it requires only $O(1)$ of extra storage when sorting n numbers. Which of the following sorting algorithm **doesn't** sort *in place*?

(A) Insertion Sort (B) Selection Sort (C) Heap Sort (D) Merge Sort

(E) Quick Sort.

8. The operation of Heap Sort is typically started by building a max-heap. Consider the array A = {4, 1, 3, 2, 6, 7, 8, 5}, which of the following is **not** the intermediate heap when building the max-heap?
   (A) {4, 6, 8, 2, 5, 7, 3, 1}　(B) {4, 1, 8, 5, 6, 7, 3, 2}
   (C) {4, 6, 8, 5, 1, 7, 3, 2}　(D) {8, 6, 7, 5, 1, 4, 3, 2}
   (E) {4, 1, 3, 5, 6, 7, 8, 2}
9. Which of the following data structure generally takes only $O(1)$ time to search a key?
   (A) Queue　(B) Priority Queue　(C) Binary Search Tree　(D) B-Tree
   (E) Hash Table.
10. Consider the following C++ class:

```
class Node {
   int n;
   int key[2];
   Node *left, *middle, *right;
   bool leaf   };
```

   Which of the following data structure is probably defined?
   (A) Binary Search Tree　(B) AVL Tree　(C) B-Tree　(D) Binomial Heap
   (E) Fibonacci Heap.
11. Which of the following data structure is often used in designing the algorithm for constructing the Huffman Codes?
   (A) Stack　(B) Queue　(C) Priority Queue　(D) Binary Search Tree
   (E) Hash Table.
12. Which of the following data structure is often used in designing the algorithm for the breadth-first search (BFS) of a graph?
   (A) Stack　(B) Queue　(C) Priority Queue　(D) Binary Search Tree
   (E) Hash Table.
13. An amortized analysis guarantees:
   (A) Average performance of each operation in the best case.
   (B) Average performance of each operation in the average case.
   (C) Average performance of each operation in the worse case.
   (D) Best performance of each operation.
   (E) Worst performance of each operation.
14. Which of the following design strategy is used in the Kruskal's algorithm for solving the minimum-spanning-tree problem?
   (A) Brute-Force　(B) Divide-and-Conquer　(C) Dynamic Programming
   (D) Greedy Algorithm　(E) Backtracking.

15. Given the following adjacency matrices, each of which represents a directed graph, which graph is strongly connected?

(A) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$　(B) $\begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$　(C) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$

(D) $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}$　(E) None of the above.

16. The Floyd-Warshall algorithm is often used to solve the all-pairs shortest-paths problem on a directed graph $G = (V, E)$. Let $d_{ij}^{(k)}$ be the weight of a shortest path from vertex i to vertex j for which all intermediate vertices are in the set {1, 2, …, k}. Then, the algorithm can be described by:

FLOYD-WARSHALL(W)
1. $n \leftarrow rows[W]$
2. $D^{(0)} \leftarrow W$
3. **for** _a_ $\leftarrow$ 1 to n
4. 　**for** _b_ $\leftarrow$ 1 to n
5. 　　**for** _c_ $\leftarrow$ 1 to n
6. 　　do $d_{i,j}^{(k)} = \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
7. Return $D^{(n)}$

Which of the following setting is the most appropriate?

(A) a $\rightarrow i$, b $\rightarrow j$, c $\rightarrow k$　(B) a $\rightarrow i$, b $\rightarrow k$, c $\rightarrow j$　(C) a $\rightarrow j$, b $\rightarrow i$, c $\rightarrow k$

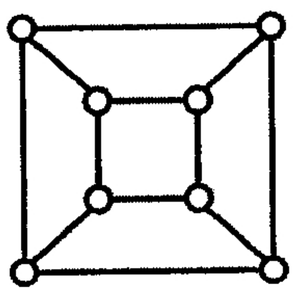
(D) a $\rightarrow k$, b $\rightarrow i$, c $\rightarrow j$　(E) None of the above.

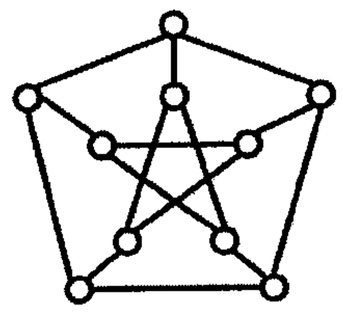
17. Following from problem (16), which of the following design strategy is used in the Floyd-Warshall algorithm?

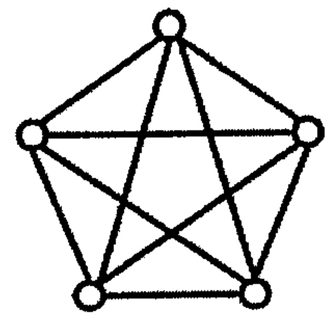
(A) Brute-Force　(B) Divide-and-Conquer　(C) Dynamic Programming

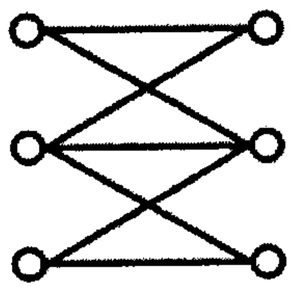
(D) Greedy Algorithm　(E) Backtracking.

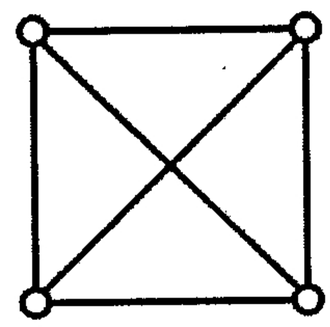
18. Which of the following graph is not a Hamiltonian?

(A) 

(B) 

(C) 

(D) 

(E) 

19. Which of the following problem is **not** a NP-Complete problem?
(A) Euler Tour (B) Hamiltonian Cycle (C) Clique (D) 3-Coloring
(E) Traveling-Salesman Problem.

20. To prove that a problem X is ***NP-Complete***, we must prove $X \in$ NP; and

(A) Select a known NP-Complete problem Y and show that $X \leq_p Y$.

(B) Select a known NP-Complete problem Y and show that $Y \leq_p X$.

(C) Select a known NP-Hard problem Y and show that $X \leq_p Y$.

(D) Select a known NP-Hard problem Y and show that $Y \leq_p X$.

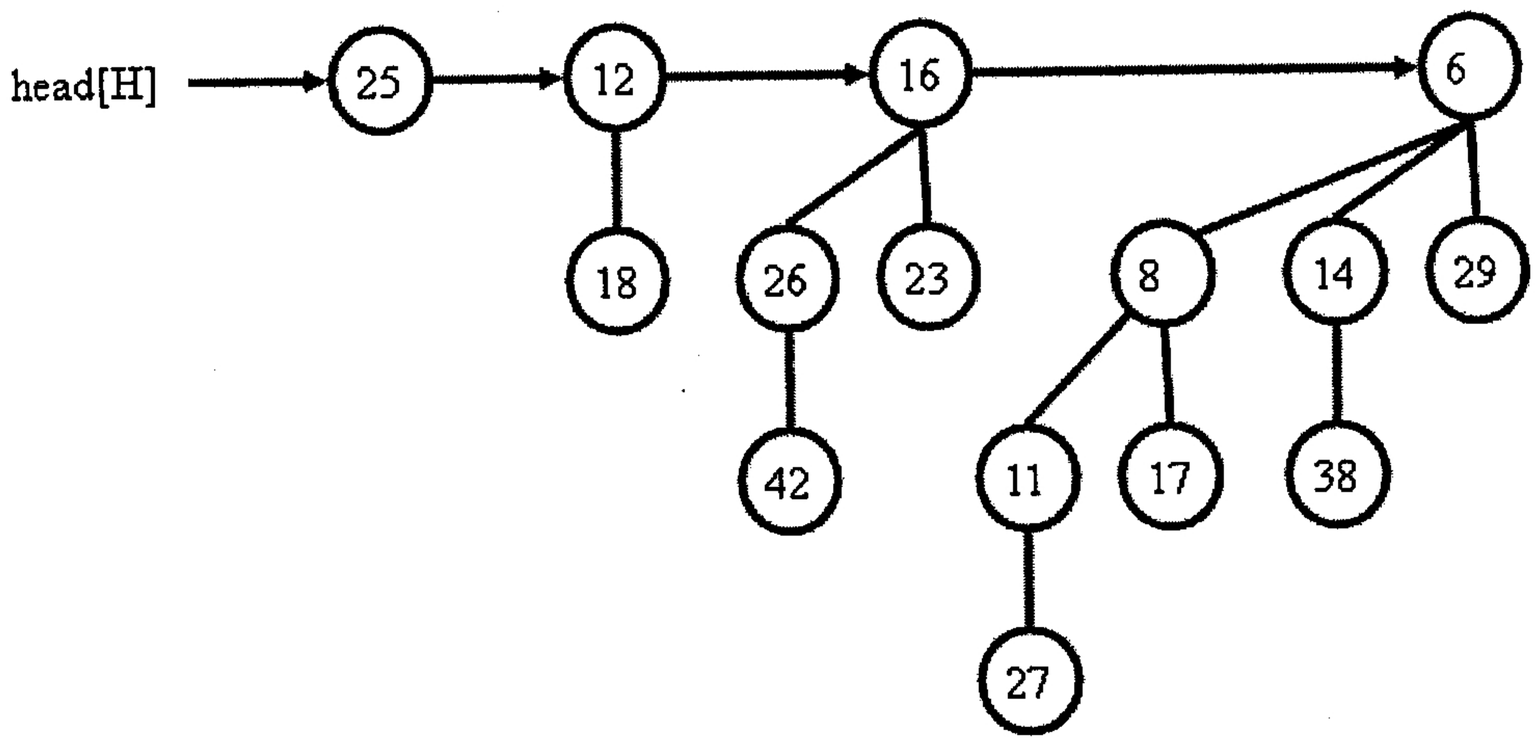
(E) None of the above.

(Note: $A \leq_p B$ means that problem A is polynomial-time reducible to problem B.)

II. **簡答題 (60%) (請寫過程，並標明答案)**

1. In software engineering, sorting algorithms are useful, but are often misused. As a software engineer, you should choose the sorting algorithm that is the most appropriate depending on the nature of the database. For each of the following databases, please suggest a sorting algorithm if *efficiency* is required assuming the number of keys to be sorted is sufficiently large. Briefly explain why. (12%)
   (a) The given data is known to be uniformly distributed over the interval [0, 1), e.g., 0.23, 0.11, 0.56, etc.
   (b) The given data may contain keys with same values. After sorting, their order in the output array must appear in the same order as they do in the input array.
   (c) The given data is known to be a banking account number with exactly 9 digits, e.g., 123456789, 111111111, 987654321, etc.
   (d) The given data is "almost" sorted with unknown distribution.

2. The following figure shows an example of a binomial heap, where the binomial heap is a collection of 4 binomial trees, each has distinct degrees. Please answer the following:
   (a) If a binomial heap is a collection of k binomial trees, each has distinct degrees, determine the total number of nodes for the binomial heap. (2%)
   (b) Illustrate the result of deleting the node 27 from the binomial heap. Please show sufficient details step by step. (8%)
   (c) What's the running time of deleting a node from the binomial heap with n nodes in asymptotic Θ-notation? (2%)

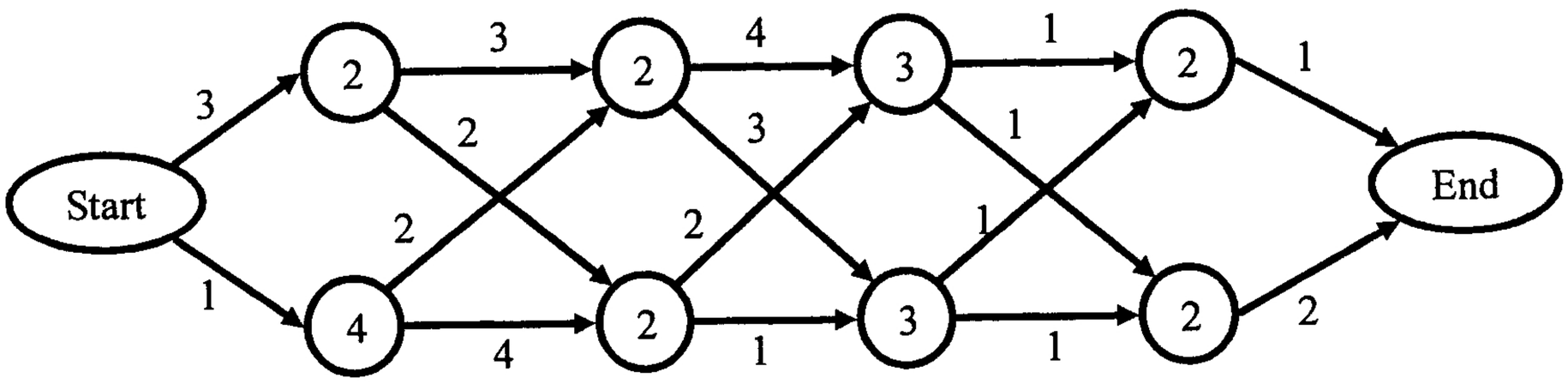


3. Given the assembly-line scheduling problem, where there are two assembly lines, each with n stations. An automobile enters the factory at the node "Start" and exits at the node "End". The time required for each station and between stations is also given. The problem is to determine the fastest way to go through the factory. Please answer the following:

(a) If the brute-force approach is used, determine the number of all possible ways to go through the factory in asymptotic Θ-notation? (2%)

(b) If the dynamic programming is used, determine the running time in asymptotic Θ-notation? (2%)

(c) Use the dynamic programming to solve the following assembly-line scheduling problem (n = 4) and determine the fastest way. Please show sufficient details (8%)

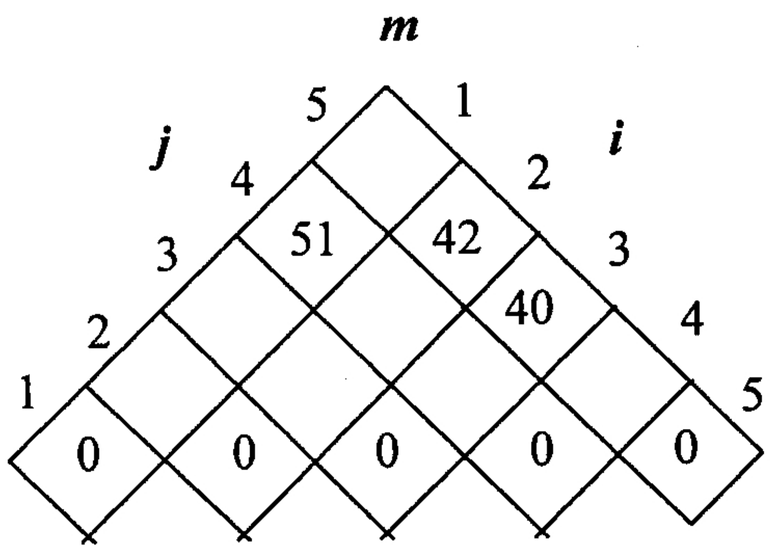


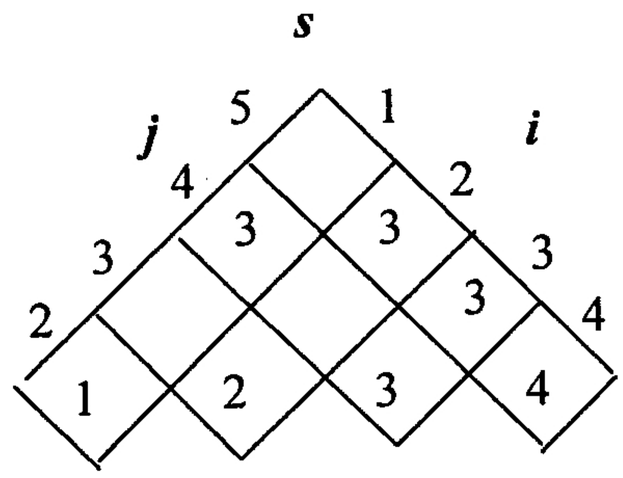
4. Given a sequence (chain) of matrices whose dimensions are listed as follows:

Matrix	A_1	A_2	A_3	A_4	A_5
Dimensions	5 × 2	2 × 3	3 × 1	1 × 7	7 × 4

We wish to compute the product $A_1A_2A_3A_4A_5$ (or $A_{1..5}$) by parenthesizing the product in a way that minimizes the number of scalar multiplications. Please solve the problem using the dynamic programming.

(a) Assume that $m[i, j]$ is the minimum number of scalar multiplications needed to compute the matrix $A_{i...j}$. In addition, let us define $s[i, j]$ to be a value at which we can split the product $A_{i...j}$ to obtain an optimal parenthesization. Please complete the given tables: (10%)





(b) Determine the optimal parenthesization. (4%)

5. For the following weighted, directed graph $G(V, E)$ with vertex set V and edge set E, use the Dijkstra's algorithm to find the single-source shortest paths for the graph. Dijkstra's algorithm maintains a set S of vertices and adds vertices in order to the set. In what order are the vertices added to the set if the source vertex is the vertex a? Please show sufficient details (10%)