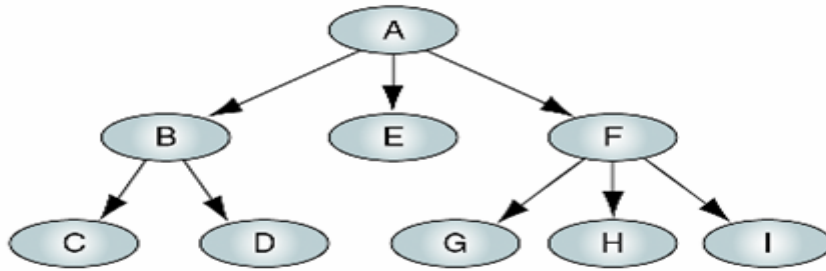


共 40 題(每題 2.5 分)

下圖是一棵 Tree 請回答 1-5 題.



- \_\_\_\_\_ 1. 橢圓的圈圈稱為 (1)Node (2)Cad (3)Seed (4)Note (5)Circle  
\_\_\_\_\_ 2. BEF 稱為 A 的 (1)Node (2)Children (3)Kids (4)Seeds (5)Roots  
\_\_\_\_\_ 3. BEF 之間的關係稱為 (1)Parent (2)Seeds (3)Roots (4)Sibling (5)Name  
\_\_\_\_\_ 4. F 稱為 GHI 的 (1)Parent (2)Seed (3)Root (4)Sibling (5)Neighbor  
\_\_\_\_\_ 5. CDEGHI 稱為 (1)Nodes (2)Seeds (3)Root (4)Sibling (5)Leaves

- \_\_\_\_\_ 6. 選出下面程式片段的複雜度 (Big-Oh)  
(1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n^4)$  (5)  $n\log(n)$

```
sum=0;
for( i=0; i<10*n; i++ )
    sum++;
for( i=0; i<10*n; i++ )
    sum++
for( i=0; i<10*n; i++ )
    sum++
for( i=0; i<10*n; i++ )
    sum++
for( i=0; i<10*n; i++ )
    sum++
for( i=0; i<10*n; i++ )
    sum++
for( i=0; i<10*n; i++ )
    sum++
```

- \_\_\_\_\_ 7. 選出下面程式片段的複雜度 (Big-Oh)  
(1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n^4)$  (5)  $n\log(n)$

```
sum=0;
for( i=0; i<n; i+=2 )
    for( j=0; j<n; j+=3 )
        sum++;
```

- \_\_\_\_\_ 8. 選出下面程式片段的複雜度 (Big-Oh)  
(1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n^4)$  (5)  $n\log(n)$

```

sum=0;
for( i=0; i<n; i++ )
    for( j=0; j<n*n; j++ )
        sum++;

```

9. 選出下面程式片段的複雜度 (Big-Oh)

- (1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n^4)$  (5)  $n \log(n)$

```

sum=0;
for( i=0; i<n; i++ )
    for( j=0; j<i*i; j++ )
        sum++;

```

10. 選出下面程式片段的複雜度 (Big-Oh)

- (1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n^4)$  (5)  $O(n^5)$

```

sum=0;
for( i=0; i<n; i++ ){
    sum = sum * 100000;
    sum = sum * 100000;
    sum *= sum + 100000;
    for( j=0; j<i; j++ )
        for( k=0; k<i+j; k++ )
            sum++;
}

```

11. 選出下面程式片段的複雜度 (Big-Oh)

- (1)  $O(n^{11})$  (2)  $O(n^{10})$  (3)  $O(n^8)$  (4)  $O(n^6)$  (5)  $n \log(n^3)$

```

sum=0;
for( i=0; i<n*n; i++ )
    for( j=0; j<i*i; j++ )
        if( j%i==0 )
            for( k=0; k<j; k++ )
                sum++;

```

12. Linked List 在實作上經常會使用 Dummy Node 的技巧，對於 Dummy Node 的敘述下列何者

- 錯誤：(1) Header 所指的第一個 Node 不放資料 (2) 第一個 Node 稱為 Dummy Node (3) Dummy Node 存放的資料是最重要的 (4) Dummy Node 使得不會有特例產生，減少程式開發的時間及 Bug 的發生

13. 請將式子  $(A+B*(C+D)/(E-F*G-H))*I-J$  轉成 前序(Prefix)表示法

- (1)  $- * + A / * B + C D - - E * F G H I J$   
 (2)  $A B C D + * E F G * - H - / + I * J -$   
 (3)  $A+B*C+D/E-F*G-H*I-J$   
 (4)  $- * + / * + - - * A B C D E F G H I J$

(5) A B C D E F G H I J+ \*\* -- / + \*-

\_\_\_\_\_14. 請用下表的 Huffmam 的編碼方式將字串 nunhund 做編碼，並列出 (b)字串編碼的結果

(1)1110010110010 (2)1101101110 (3)001010101 (4) 10101010 (5)  
0100110100111

字元	d	h	n	u
Huffman 碼	111	110	0	10

\_\_\_\_\_15. 下列排序方法中，以那一種的平均速度最快：(1)泡沫排序 (2)選擇排序 (3)快速排序  
(4)插入排序。

\_\_\_\_\_16. 以循序搜尋法搜尋某一資料，最多要搜尋幾次，才能從 1000 筆資料中找到所要的資料  
(1) 1 (2)10 (3)11 (4)1000

\_\_\_\_\_17. 以二分搜尋法搜尋某一資料，最多要搜尋幾次，才能從 1000 筆資料中找到所要的資料？(1)  
1 (2)10 (3)11 (4)1000

\_\_\_\_\_18. FILO(First In Last Out)屬於資料結構的那一種？(1)Queue (2)List (3)Tree  
(4)Stack

\_\_\_\_\_19. 資料量很大的排序，通常必須使用到輔助記憶體，這種排序稱為：(1)外部排序 (2)內部  
排序 (3)快速排序 (4)插入排序

\_\_\_\_\_20. 有一場演唱會，票面上註記 4 位數號碼，千位數表示等級，有三種等級(1, 2, 3)，3 表示最  
高等級擁有最優先入場權力，其餘三位數表示序號，由 999 排至 000，999 表示所賣出第一  
張票，擁有該等級票的最優先入場權力。請問演唱會開演前有下列五張票號依續到達，請問  
在開放入場後，這些票號入場的順序？

到達順序	票號
1	1234
2	3986
3	3876
4	1387
5	2876

(1) 3986, 3876, 2876, 1387, 1234 (2) 1387, 1234, 3986, 3876, 2876  
(3) 1234, 3986, 3876, 1387, 2876 (4) 1234, 3986, 3876, 2876, 1387  
(5) 3876, 2876, 1387, 1234, 3986

\_\_\_\_\_21. 下列那一種四則運算的表示法，較合適電腦來直接執行計算？(1) 前序(prefix)表示法 (2)  
中序(infix)表示法 (3) 後序(postfix)表示法 (4) 以上皆是

- \_\_\_\_\_22. 下列那一種不是樹的 traversal? (1) inorder traversal (2) preorder traversal (3) postorder traversal (4) complete traversal
- \_\_\_\_\_23.  $n$  個節點(node)的連接樹(connected tree)擁有幾條線(edge)? (1)  $n-2$  (2)  $n-1$  (3)  $n$  (4)  $n+1$
- \_\_\_\_\_24. 二元樹(binary tree)每個節點(node)不可能擁有的子女(Children)數為多少? (1) 0 (2) 1 (3) 2 (4) 3
- \_\_\_\_\_25. 高度(height)為 4 的二元樹(binary tree)最多有幾個節點(node)? (1) 15 (2) 16 (3) 17 (4) 18
- \_\_\_\_\_26. 高度(height)為 5 的二元樹(binary tree)最少有幾個節點(node)? (1) 4 (2) 5 (3) 8 (4) 16
- \_\_\_\_\_27. 二元搜尋樹(binary search tree)不擁有以下那一個功能? (1) 搜尋一個特定值 (2) 插入一個特定值 (3) 刪除一個特定值 (4) 改變一個節點的值
- \_\_\_\_\_28. 一個圖(graph)可以用那一種資料結構表示? (1) adjacency matrix (2) adjacent list (3) adjacent multilist (4) 以上皆是
- \_\_\_\_\_29. 插入排序法(insertion sort)的時間複雜度(Big-Oh)是多少? (1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n \log n)$
- \_\_\_\_\_30. 快速排序法(quick sort)的時間複雜度(Big-Oh)是多少? (1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n \log n)$
- \_\_\_\_\_31. 合併排序法(merge sort)的時間複雜度(Big-Oh)是多少? (1)  $O(n)$  (2)  $O(n^2)$  (3)  $O(n^3)$  (4)  $O(n \log n)$
- \_\_\_\_\_32. Kruskal 的演算法可找到一個生成樹(spanning tree), 下列那一項不是這棵樹的特性? (1) 最小成本(cost) (2) 沒有迴圈(cycle) (3) 是一個連接(connected)的子圖 (4) 節點(vertex)和線(edge)的數目相同
- \_\_\_\_\_33. 鏈節串列(Linked List)的優點是什麼? (1) 不需佔據連續的記憶體 (2) 不需用到指標 (3) 資料彼此隨意連接也不會消失 (4) 尋找資料時比矩陣快
- \_\_\_\_\_34. 解決 5 個碟子的河內塔(Hanoi Tower)問題需要搬動幾次碟子? (1) 15 (2) 31 (3) 63 (4) 16
- \_\_\_\_\_35. 兩個  $m \times n$  的整數矩陣相加需要執行幾次整數加法? (1)  $m+n$  (2)  $m \times n$  (3)  $m \times m$  (4)  $n \times n$

- \_\_\_\_\_36. 堆疊(stack)在輸出資料時是從何處取得? (1) 頂端 (2) 底部 (3) 任意處 (4) 不可讀取只可寫入
- \_\_\_\_\_37. 佇列(queue)在輸出資料時是從何處取得? (1) 頭部 (2) 尾巴 (3) 任意處 (4) 不可讀取只可寫入
- \_\_\_\_\_38. 堆疊(stack)在寫入資料時是從何處輸入? (1) 頂端 (2) 底部 (3) 任意處 (4) 不可寫入只可讀取
- \_\_\_\_\_39. 佇列(queue)在寫入資料時是從何處輸入? (1) 頭部 (2) 尾巴 (3) 任意處 (4) 不可寫入只可讀取
- \_\_\_\_\_40. 下列哪一個演算法其輸入資料結構必須是一棵樹 (tree)? (1) shortest path (2) connected components (3) depth-first search (4) postorder traversal