

# 《計算機概要》

- B** 1 BCD數元(0101 1001 0111)<sub>BCD</sub>轉換成 16 進制後其值為何?  
 (A)(597)<sub>16</sub> (B)(255)<sub>16</sub> (C)(010110010111)<sub>16</sub> (D)(1125)<sub>16</sub>
- D** 2 下列四種數值資料型別 (data type)，何者可表示的數值資料範圍最大?  
 (A)整數 (integer) (B)長整數 (long) (C)單精度 (single) (D)倍精度 (double)
- A** 3 下列何者不屬於程序間通訊 (interprocess communication) 方式?  
 (A)中斷 (interrupt) (B)共享記憶體 (shared memory)  
 (C)訊息傳遞 (message passing) (D)插座 (sockets)
- B** 4 假設記憶體中儲存一整數 (Integer) 資料必須使用 4 位元組 (Byte)，若 C 語言程式中宣告一整數陣列 (Array) 為 m[20] 且 m[11] 儲存於記憶體中之位址 (Address) 為 610620，則下列各選項何者正確?  
 (A)陣列 m 共計使用 84 位元組 (Byte) 之記憶體儲存空間  
 (B) m[1] 儲存於記憶體中之位址 (Address) 為 610580  
 (C) m[19] 儲存於記憶體中之位址 (Address) 為 610656  
 (D)記憶體位址 (Address) 為 610633 之字元被配置用於儲存 m[13]
- B** 5 關於排序演算法之運算時間複雜度的敘述，何者正確?  
 (A)無論資料的初始排列情況如何，快速排序法 (quick sort) 的運算時間複雜度總是  $\theta(n \log n)$   
 (B)無論資料的初始排列情況如何，合併排序法 (merge sort) 的運算時間複雜度總是  $\theta(n \log n)$   
 (C)就平均運算時間複雜度而言，氣泡排序法 (bubble sort) 優於合併排序法  
 (D)就最差運算時間複雜度而言，快速排序法優於氣泡排序法
- A** 6 將 n 筆資料插入到一個雜湊表 (hash table) 結構中，假設沒有碰撞 (collision) 產生，令搜尋任一筆資料所花費的時間為 T，則：  
 (A)  $T = \theta(1)$  (B)  $T = \theta(n)$  (C)  $T = \theta(\log n)$  (D)  $T = \theta(n \log n)$
- C** 7 以下有關 Java 的敘述，何者錯誤?  
 (A)是一種物件導向語言  
 (B) Java 程式的執行速度通常較 C 程式慢  
 (C) Java 與 C++ 均能處理指標型態  
 (D)只要有 Java virtual machine 的瀏覽器，即可執行 Java 編譯的執行碼
- C** 8 以下何種協定屬於 OSI 7 層協定的應用層 (Application layer)?  
 (A) ICMP (B) ARP (C) SMTP (D) UDP
- B** 9 一個 5 節點 (nodes) 的無方向性 (undirected) 完全圖 (complete graph) 共有幾個邊 (edges)?  
 (A) 5 (B) 10 (C) 20 (D) 25
- A** 10 在使用 log-based recovery 的資料庫系統中，當系統當機後，會使用 log 來做為系統回復的依據；下面是一段系統的記錄，根據這段記錄所做的系統回復動作，下列敘述何者正確?  
 <T<sub>0</sub> start>  
 < T<sub>0</sub>, A, 1000, 950>  
 < T<sub>0</sub>, B, 2000, 2050>  
 < T<sub>0</sub> commit>  
 <T<sub>1</sub> start>  
 < T<sub>1</sub>, C, 700, 600>  
 (A)完成系統回復後，C 的值為 700 (B)完成系統回復後，B 的值為 2000  
 (C)完成系統回復後，A 的值為 1000 (D)系統回復後，會做undo(T<sub>0</sub>)的動作

高點·高上高普特考 [goldensun.get.com.tw](http://goldensun.get.com.tw) 台北市開封街一段 2 號 8 樓 02-23318268

【中壢】中壢市中山路 100 號 14 樓·03-4256899

【台中】台中市東區復興路四段 231-3 號 1 樓·04-22298699

【台南】台南市中西區中山路 147 號 3 樓之 1·06-2235868

【高雄】高雄市新興區中山一路 308 號 8 樓·07-2358996

【另有板橋·淡水·三峽·林口·羅東·逢甲·東海·中技·雲林·彰化·嘉義】

- A 11 以下兩個 C 語言的指令  $x=2; y=x--$  ; 被執行過後,  $x$  與  $y$  的值分別為何?  
 (A) 1 與 2 (B) 1 與 1 (C) 2 與 1 (D) 2 與 0

- D 12 下列以 C 程式語言撰寫之程式執行後產生之輸出為何?

```
#include <stdio.h>
int main() {
    int choice = 1;
    switch (choice)
    {
        case 1:
            printf("*");
        case 2:
            printf("***");
        case 3:
            printf("*****");
    }
    return 0;
}
```

- (A)\* (B)\*\* (C)\*\*\* (D)\*\*\*\*\*

- A 13 已知函數  $F(x, y, z) = xz + y\bar{z}$ , 試問下列何者為此函數  $F$  之最大項之積 (product of maxterms) ?  
 (A)  $(x + y + z)(x + y + \bar{z})(x + \bar{y} + \bar{z})(\bar{x} + y + z)$  (B)  $(x + \bar{y} + z)(\bar{x} + y + \bar{z})(\bar{x} + \bar{y} + z)(\bar{x} + \bar{y} + \bar{z})$   
 (C)  $(x + y + z)(x + y + \bar{z})(x + \bar{y} + z)(\bar{x} + y + \bar{z})$  (D)  $(x + \bar{y} + \bar{z})(\bar{x} + y + z)(\bar{x} + \bar{y} + z)(\bar{x} + \bar{y} + \bar{z})$

- C 14 下圖是布林函數  $F(x, y, z)$  的 K-map, 關於  $F(x, y, z)$  的表示法何者錯誤?

	$x$	$yz$	00	01	11	10
0	0	1	0	1	0	1
1	0	1	1	0	1	0

- (A)  $F(x, y, z) = x \oplus y \oplus z$  (B)  $F(x, y, z) = \sum(1, 2, 4, 7)$   
 (C)  $F(x, y, z) = \sum(1, 3, 4, 6)$  (D)  $F(x, y, z) = \prod(0, 3, 5, 6)$

- C 15 下列何項機制使得記憶體與 I/O 裝置進行資料傳輸時, 資料可以無需透過 CPU 而由 I/O 裝置寫入到記憶體?  
 (A) Programmed I/O (B) Interrupt-driven I/O (C) DMA (D) Isolated I/O

- A 16 下表為一位元全減法器 (one-bit full subtractor) 之真值表, 其中 X 為被減數, Y 為減數, Z 為次一級較低權重借位值。試問 (a, b, c) 值各為多少?

全減法器真值表

輸入			輸出	
X	Y	Z	借位 (B)	差值 (D)
0	0	0	0	0
0	0	1	a	1
0	1	0	1	1
0	1	1	1	0
-----				
1	0	0	b	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	c

- (A)(a, b, c)=(1, 0, 1) (B)(a, b, c)=(0, 0, 1) (C)(a, b, c)=(1, 0, 0) (D)(a, b, c)=(0, 1, 1)

- D 17 假設有五個程序（甲、乙、丙、丁、戊）同時送入電腦執行，它們的執行時間分別是 5、4、3、2、1 分鐘，如果該電腦是以甲、乙、丙、丁、戊的順序來循序且不經打斷地執行，請問該五個程序的平均回轉時間（Turnaround Time）是多少分鐘？  
 (A) 3 分鐘 (B) 5 分鐘 (C) 8 分鐘 (D) 11 分鐘
- A 18 假設 CPU 的時脈是 2GHz，某程式執行  $10^{10}$  道指令花了 6 秒，則該程式在此一部分的 CPI (cycles per instruction) 等於多少？  
 (A) 1.2 (B) 1.5 (C) 2.0 (D) 3.0
- C 19 下列三種二進位的整數表示法：符號-大小 (Sign-magnitude)、1 的補數 (1's Complement) 以及 2 的補數 (2's Complement) 對十進位數“-117”的表示何者完全正確？  
 (A) Sign-magnitude:11101101; 1's Complement:10010010; 2's Complement:10010011  
 (B) Sign-magnitude:11110101; 1's Complement:10001011; 2's Complement:10001010  
 (C) Sign-magnitude:11110101; 1's Complement:10001010; 2's Complement:10001011  
 (D) Sign-magnitude:10010010; 1's Complement:11101101; 2's Complement:10010011
- D 20 令整數資料以 8 位元及 2 的補數 (2's complement) 表示，且 x、y 和 z 的二進制位元表示分別是  $x=1100\ 1000$ ， $y=1100\ 1011$ ， $z=1000\ 0110$ 。在執行 2 的補數 (2's complement) 運算時，下列何者錯誤？  
 (A)  $-x$  (負 x) = 0011 1000 (B)  $-y=0011\ 0101$  (C)  $x-y=1111\ 1101$  (D)  $x+z=1100\ 1110$
- C 21 下列那一個指令可以依需要來改變 PC (program counter) 的值？  
 (A) AND (B) MOV (C) JPNZ (D) NOR
- B 22 「圖形 (graph) 中的每一個節點 (node) 與其它所有節點都相鄰 (adjacent)」為下列那一種圖形的定義？  
 (A) 獨立圖形 (isolated graph) (B) 完整圖形 (complete graph)  
 (C) 有限圖形 (finite graph) (D) 緊密連通圖形 (strongly connected graph)
- B 23 針對鏈結串列 (linked list)，以下敘述何者正確？  
 (A) 指定整數 k，能在  $O(1)$  時間完成存取鏈結串列中的第 k 個節點  
 (B) 給定一個節點 x，能在  $O(1)$  時間完成插入新元素到節點 x 之後  
 (C) 由於使用指標不會浪費許多記憶體空間，因此鏈結串列很適合用在靜態的資料結構中  
 (D) 指定一個資料 value，能在  $O(1)$  時間找到串列中儲存資料 value 的節點
- D 24 假設二維陣列 char A[N][N] 採用以列為主的順序 (row major order) 而且索引值從 0 開始，則元素  $A[i][j]$  儲存於那一個記憶體位址？  
 (A)  $A+i*j$  (B)  $A+i+j*N$  (C)  $A+(i+j)*N$  (D)  $A+i*N+j$
- B 25 以下何種應用較不適合使用堆疊 (stack) 來解決？  
 (A) 解決迷宮 (maze) 問題時用來記錄走過的路徑  
 (B) 圖形 (graph) 的廣度優先搜尋 (breadth-first search)  
 (C) 運算式的前置 (prefix) 與後置 (postfix) 表示法的轉換  
 (D) 樹狀結構的中序走訪 (inorder traversal)
- B 26 以下對於佇列 (Queue) 資料型態的敘述何者正確？  
 (A) 是一種無序的串列 (List)  
 (B) 插入與刪除元素的動作發生在佇列的不同端  
 (C) 是一種後進先出 (Last-In First-Out) 的串列  
 (D) 基本的操作 (Operations) 包括插入、刪除與隨機存取佇列元素

- C 27 下列以 C 程式語言撰寫之程式執行後產生之輸出為何？

```
#include <stdio.h>
int main() {
    int a = 10, b = 5, c = 1, d = 7;

    if (!a > 100 && b < 6) {
        if (c < 4 || d > 8)
            printf("A");
        else
            printf("B");
    }
    else
        printf("C");
    return 0;
}
```

- (A) A (B) B (C) C (D) 無任何輸出

- C 28 下列以 C 程式語言撰寫之程式執行後產生之輸出為何？

```
#include <stdio.h>
int main()
{
    int a = 64, b = 1;
    printf("%d", a >> (1+(b<<2)));
    return 0;
}
```

- (A) 0 (B) 1 (C) 2 (D) 4

- A 29 下列那一項是 NAT (Network Address Translation) 的主要功能？

- (A) 讓多個主機 (host) 共享一個公共 IP 位址 (B) 將 IP 位址轉換成主機名稱 (host name)  
(C) 將主機名稱 (host name) 轉換成 IP 位址 (D) 將 IP 位址轉換成 Port 號碼

- C 30 關於線路交換 (Circuit Switching) 與封包交換 (Packet Switching)，以下敘述何者正確？

- (A) 線路交換在連線時建立實體線路，多組用戶以多工方式共用此線路  
(B) 線路交換採用 Datagram 技術傳送封包  
(C) 封包交換比線路交換能容納更多組用戶同時進行連線  
(D) 若採用線路交換技術，頻寬不足時可以動態在多組線路之間切換

- C 31 下列 SQL 語法何者錯誤？

- (A) select count(\*) from students  
(B) select MAX(age) from students  
(C) select name, MAX(age) from students  
(D) select name, MAX(age) from students  
group by name

- D 32 下列何者不是現有資料庫軟體常用的索引結構？

- (A) B<sup>+</sup>樹 (B<sup>+</sup> tree) (B) R樹 (R tree)  
(C) 雜湊 (Hashing) (D) 二元搜尋樹 (Binary search tree)

- B 33 下列中序 (Infix) 運算式之前序 (Prefix) 運算式表示方式為何？

$(A+B \times C) \times (A/B - C) / (A \times B + B \times C + C \times A)$

- (A)  $\times / + \times ABC / - ABC + \times AB + \times BC \times CA$  (B)  $\times + A \times BC - / ABC + \times AB \times BC \times CA$   
(C)  $/ \times + \times BCA - / ABC + \times AB + \times BC \times CA$  (D) 以上皆非

【中壢】中壢市中山路 100 號 14 樓 · 03-4256899

【台中】台中市東區復興路四段 231-3 號 1 樓 · 04-22298699

【台南】台南市中西區中山路 147 號 3 樓之 1 · 06-2235868

【高雄】高雄市新興區中山一路 308 號 8 樓 · 07-2358996

【另有板橋 · 淡水 · 三峽 · 林口 · 羅東 · 逢甲 · 東海 · 中技 · 雲林 · 彰化 · 嘉義】

A 34 某影片使用壓縮技術，將資料速率 (data rate) 降低為 3 Mbits/sec，影片以每秒 40 個畫面 (frame)、每個畫面以  $800 \times 600$  解析度播放，且每一個畫素 (pixel) 以 24 位元表示，則壓縮率的最小值為何，才能滿足上述的規格？

- (A) 153.6:1                      (B) 128.7:1                      (C) 192.3:1                      (D) 238.4:1

B 35 若  $G$  為一非多重圖形 (non-multigraph)、無自身邊線 (Self edge) 之無向圖形 (Undirected graph) 結構，並以  $n_G$  表示  $G$  之頂點 (Vertex) 數，以  $e_G$  表示  $G$  之邊線 (Edge) 數，且  $T_G$  為基於  $G$  之生成樹 (Spanning tree)。下列為有關  $G$  與其生成樹  $T_G$  之敘述：

- ① 生成樹  $T_G$  可經由對  $G$  使用 Kruskal 演算法或 Prim 演算法產生。
- ② 若以  $n_T$  表示生成樹  $T_G$  之節點 (Node) 數，則  $n_T = n_G$ 。
- ③ 若以  $e_T$  表示生成樹  $T_G$  之邊線 (Edge) 數，則  $e_T < e_G$ 。
- ④ 若以  $h_T$  表示生成樹  $T_G$  之高度 (Height)，則  $\log_2 n_G \leq h_T \leq n_G$ 。〔註：僅有樹根 (Root) 節點之樹狀 (Tree) 結構其高度為 1。〕
- ⑤ 若  $T_G$  為基於  $G$  之唯一生成樹 (Spanning tree)，則  $G$  為一樹狀 (Tree) 結構。

請選出最適合之選項：

- (A) ②③正確；①⑤錯誤                      (B) ①②正確；③④錯誤  
 (C) ①④錯誤                                      (D) ②④正確

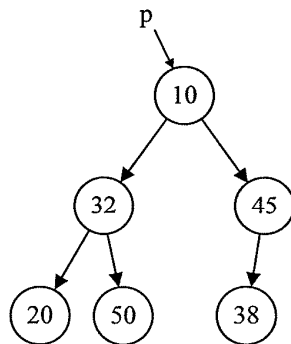
A 36 下列何者不是布林函數 (Boolean function)  $F(A, B, C, D) = \sum m(0, 2, 3, 5, 6, 7, 8, 10, 14, 15)$  必要的主要項 (essential prime implicant)？

- (A)  $CD'$                                       (B)  $BC$                                       (C)  $A'C$                                       (D)  $A'BD$

D 37 樹 (tree) 中每一節點有 data、left、right 三個欄位。data 儲存資料而 left 和 right 兩個指標分別指向左子樹和右子樹。函式 print() 如下：

```
void print(node* x) {
    cout << x->data;
    print(x->left);
    print(x->right);
}
```

針對以下的樹，呼叫 print(p) 結果為何？



- (A) 10、32、45、20、50、38                      (B) 20、32、50、10、38、45  
 (C) 32、20、50、45、38、10                      (D) 10、32、20、50、45、38

B 38 下列以 C 程式語言撰寫之程式執行後的輸出為何？

```
#include <stdio.h>
int foo1(int v)
{
    int a = 0;
    a += v;
    return a;
}
int foo2(int v)
{
    static int a = 0;
    a += v;
    return a;
}
int main ()
{
    foo1(10);
    foo2(10);
    printf("%d %d", foo1(5), foo2(5));
    return 0;
}
```

(A) 5 5

(B) 5 15

(C) 15 5

(D) 15 15

A 39 下列以 C 程式語言撰寫之程式執行後產生之輸出為何？

```
#include <stdio.h>
int gcd(int x, int y)
{
    if (x%y == 0) return y;
    else return gcd(y, x%y);
}
int main()
{
    int x,y;
    printf("%d", gcd(231, 35));
    return 0;
}
```

(A) 7

(B) 21

(C) 231

(D) 35

B 40 以下敘述何者錯誤？

- (A) 傳訊人使用其私密金鑰 (private key) 將原始訊息的摘要 (digest) 進行加密，即得到此訊息的數位簽章 (digital signature)
- (B) 數位信封 (digital envelop) 的觀念就是使用收訊人的私密金鑰 (private key) 對某些機密資料作加密，收訊人收到後再使用自己的公開金鑰 (public key) 解密而讀取資料
- (C) SSL (Secure Socket Layer) 架構於 TCP 傳輸層 (TCP transport layer) 之上，應用層 (application layer) 之下
- (D) 數位憑證 (digital certificate) 一般由憑證機構 (certification authority, CA) 發出，內含有憑證持有者的公開金鑰 (public key)