

# 臺北市立第一女子高級中學 106 學年度資訊能力競賽 複賽筆試試題詳解(v.03)

2017/09/26

一、 $(101100000000)_2$ 、 $(2845)_{10}$ 、 $(5436)_8$ 、 $(B1F)_{16}$ ，這 4 個數字中，哪一個數值最大？\_\_\_\_\_ (1) \_\_\_\_\_。

解答:

$$(101100000000)_2 = (B00)_{16}$$

$$(2845)_{10} = (B1D)_{16}$$

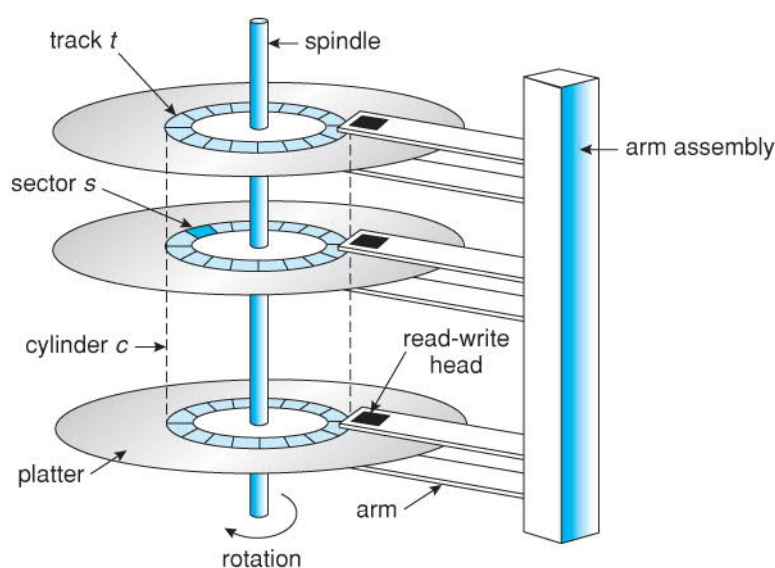
$$(5436)_8 = (101100011110)_2 = (B1E)_{16}$$

因此，答案為 $(B1F)_{16}$ 。

二、某硬碟上標示「C/H/S：19328/16/64」，若已知格式化後每個磁區有 512Bytes，請問此硬碟的總容量為多少 GB？\_\_\_\_\_ (2) \_\_\_\_\_。

解答:

C/H/S 分別為 Cylinder/Head/Sector 之意，亦即“磁柱/讀寫頭/磁區”，如下圖。



本題的硬碟有 19328 個磁柱、16 個讀寫頭(8 片磁片, 16 個讀寫面)、每一個讀寫面上的每一個磁柱上有 16 個磁區。所求為

$$\frac{19328 \times 16 \times 64 \times 512}{1024 \times 1024 \times 1024} = 9.4375$$

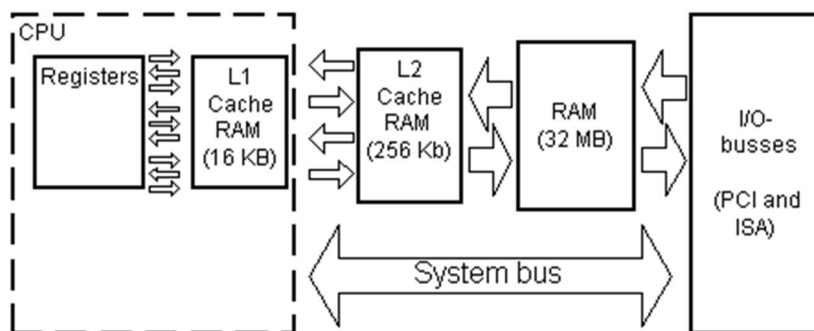
故答案為 9.4375GB。

三、請依存取速度由快至慢，排序下列記憶體類型(A)DRAM、(B)SRAM、(C)暫存器、(D)L1快取記憶體？\_\_\_\_\_ (3)。

解答:

本題中的 4 種記憶體，其距離 CPU(中央處理單元)愈近者，存取速度愈快。

而「暫存器(register)」位於 CPU 內部的 ALU(算術邏輯單元)之中、「L1 快取記憶體(L1 Cache)」的位置介於 ALU 與「L2 快取記憶體(由 SRAM 組成)」之間、「L2 快取記憶體」又介於「L1 快取記憶體」與「主記憶體(由 DRAM 組成)」之間，如下圖。



所謂的 DRAM，全名為 Dynamic Random Access Memory。相對於 DRAM, SRAM 的 S 代表 Static。DRAM 在保持記憶內容時，需要時時補充電力，因此存取速度比不需要充電的 SRAM 來得慢，但價格比 SRAM 便宜許多。

因此，答案為 CDBA。

四、哪一種儲存裝置是使用快閃記憶體製成，具有耐震、低耗電等優點，可用於取代傳統硬碟或與傳統硬碟搭配使用？\_\_\_\_\_ (4)。

解答:

固態硬碟（英語：Solid State Disk、Solid State Drive，簡稱 SSD）是一種以快閃記憶體作為永久性記憶體的電腦儲存裝置。SSD 不使用「碟片」來記存資料，其內部完全為電子裝置、沒有類似「馬達」、「存取臂」、「讀寫頭」等機械裝置，於是具有耐震與低耗電等優點。

因此，答案為 SSD。

五、小綠想將自己的創作歌曲燒錄至 CD 中並寄至選秀節目參加甄選，請問她應用甚麼軟體進行燒錄？(A)WinRAR、(B)CDBurnerXP、(C)GOM Player、(D)ACDSee\_\_\_\_\_ (5)。

解答:

WinRAR 為著名的壓縮/解壓縮軟體

CDBurnerXP 為光碟燒錄軟體

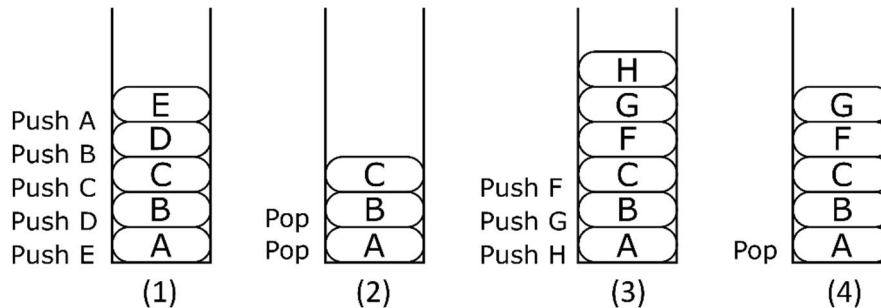
GOM Player 為媒體播放軟體

ACDSee 為圖片瀏覽軟體

故答案為 B

六、使用堆疊的資料結構，依序放入(push) A,B,C,D,E 五個字母元素，再拿出(pop)兩個元素後，再放入(push) F,G,H 三個元素，最後拿出(pop)一個元素，請問堆疊剩餘的元素，由上而下依序為？\_\_\_\_\_ (6)。

解答:



故答案為 GFCBA。

七、哪一種硬體裝置，可以管制資料的流向，並限定外界能存取內部網路的權限，以保障內部網路不會被人輕易入侵？\_\_\_\_\_ (7)。

解答:

在電腦運算領域中，防火牆(Firewall)是一項協助確保資訊安全的裝置，會依照特定的規則，允許或是限制傳輸的資料通過。防火牆可能是一台專屬的硬體或是架設在一般硬體上。  
(引用自維基百科)

故答案為“防火牆”。

八、有一後序表示運算式  $52 * 4 + 63 * -78 * +$ ，數字為個位數，請問運算結果為？\_\_\_\_\_ (8)。

解答:

$$\begin{aligned}
 & 52 * 4 + 63 * -78 * + \\
 & = (5)(2) * (4) + (6)(3) * -(7)(8) * + \\
 & = (10)(4) + (6)(3) * -(7)(8) * + \\
 & = (14)(6)(3) * -(7)(8) * + \\
 & = (14)(18) - (7)(8) * + \\
 & = (14)(18) - (56) + \\
 & = (-4)(56) + \\
 & = 52
 \end{aligned}$$

故答案為 52。

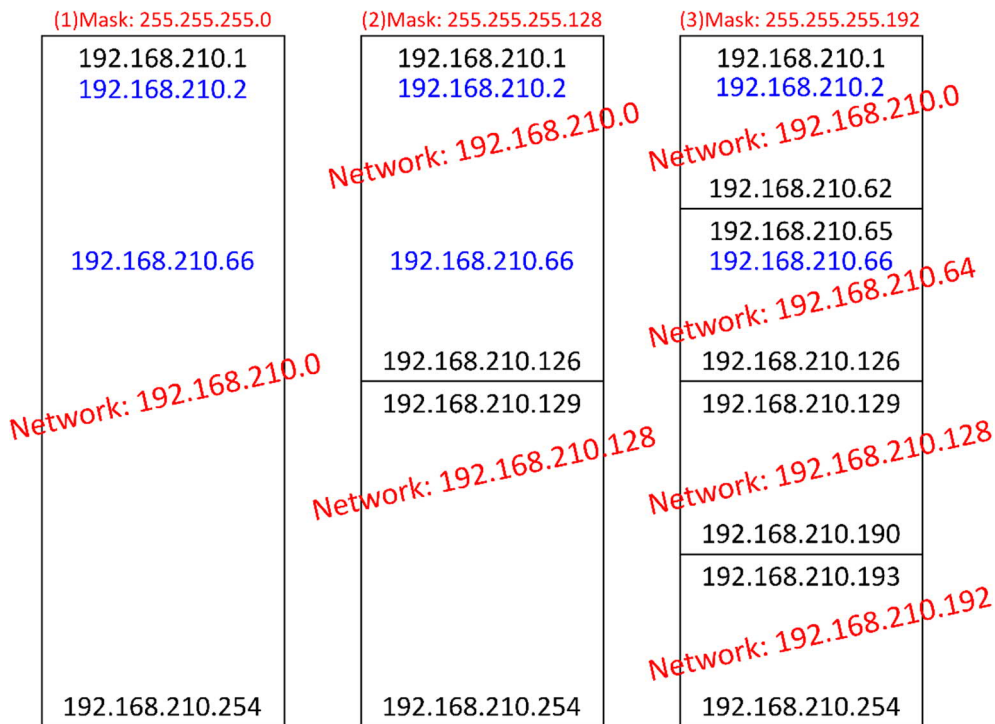
九、可使 IP 位址 192.168.210.2 和 192.168.210.66 同屬於一個子網路，且該子網路的 IP 位址數為最小的子網路遮罩為？\_\_\_\_\_ (9) 。

解答：

192.168.210.2 與 192.168.210.66 的前 3 個數字(192.168.210)相同，故可得知遮罩 255.255.255.0 可使此 2 組 IP 位址同屬網路位址為 192.168.210.0 的子網路，而此子網路可容納 254 個 IP 位址。

若將 192.168.210.0 網路切為 2 個子網路，則網路遮罩可設定為 255.255.255.128，此 2 子網路的網路位址將為 192.168.210.0 與 192.168.210.128，每個子網路內可容納 126 個 IP 位址。而且，192.168.210.2 與 192.168.210.66 均同屬於 192.168.210.0 子網路。

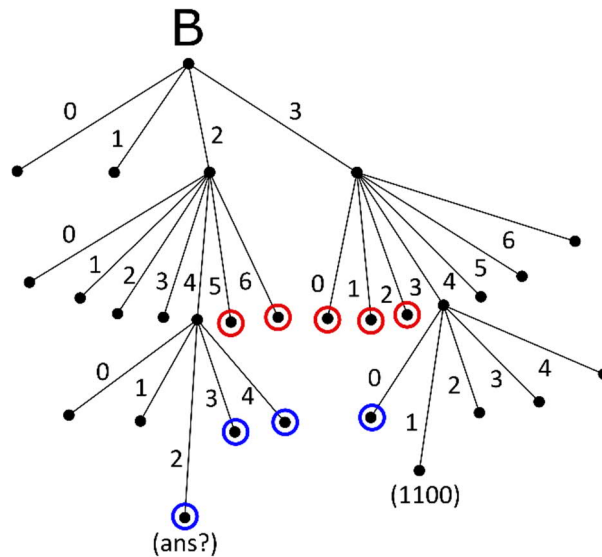
若將 192.168.210.0 網路切為 4 個子網路，則網路遮罩可設定為 255.255.255.192，此 4 子網路的網路位址將分別為 192.168.210.0、192.168.210.64、192.168.210.128、192.168.210.192。而 192.168.210.2 與 192.168.210.66 將因此分別屬於 192.168.210.0 子網路與 192.168.210.64 等 2 個不同的子網路。(如下圖)



故答案為 255.255.255.128。

十、一個三維陣列 B[0~3,0~6,0~4] 用來存放整數，假設每一個陣列元素包含兩個記憶體位址，以 row-major 排列，若 B[3][3][1] 開始於記憶體位置 1100，請問 B[2][4][2] 於記憶體位置為何？\_\_\_\_\_ (10)。

解答:



從上圖中，可知 B[2][5][0] 到 B[3][2][4] 共有 25 個節點，加上 B[2][4][2]、B[2][4][3]、B[2][4][4]、B[3][3][0] 等 4 個節點，共有 29 個節點。而每個節點使用 2 個記憶體位址，因此，這些節點總共使用了  $29 \times 2 = 58$  個記憶體空間。

已知 B[3][3][1] 的位置為 1100，B[2][4][2] 的記憶體位址則為  $1100 - 58 = 1042$ ，

故答案為 1042。

十一、假設某二元樹的中序走訪 (inorder traversal) 為 FDHGIBEAC，前序走訪 (preorder traversal) 為 ABDFGHIEC，則此二元樹的後序走訪 (postorder traversal) 為？\_\_\_\_\_ (11)。

解答:

二元樹的「前序走訪 (preorder traversal)」演算法如下:

```
void preorder(Node* p)
{
    if (p!=NULL)
    {
        cout << p->data;
        preorder(p->left);
        preorder(p->right);
    }
}
```

而且，二元樹的「中序走訪 (inorder traversal)」演算法如下:

```

void inorder(Node* p)
{
    if (p!=NULL)
    {
        inorder(p->left);
        cout << p->data;
        inorder(p->right);
    }
}

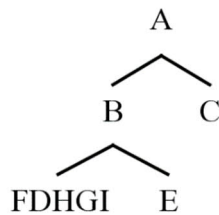
```

所以二元樹的「前序走訪」次序可以用來分割「中序走訪」的結果，如下：

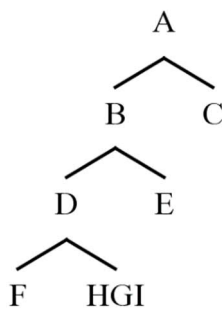
- (1) 前序走訪的第 1 個字母 'A'，可以把中序走訪的序列 "FDHGIBEAC" 分割為 "FDHGIBE" 與 "C"，如下圖。



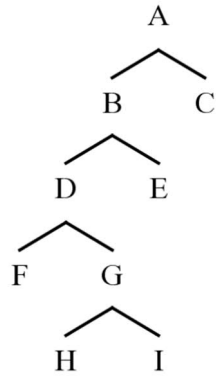
- (2) 前序走訪的第 2 個字母 'B'，可以把中序走訪的序列 "FDHGIBE" 分割為 "FDHGI" 與 "E"，如下圖。



- (3) 前序走訪的第 3 個字母 'D'，可以把中序走訪的序列 "FDHGI" 分割為 "F" 與 "HGI"，如下圖。



- (4) 前序走訪的第 4 個字母 'F'，由上圖可知已為終端節點。因此，第 5 個字母 'G' 可以把中序走訪的序列 "HGI" 分割為 "H" 與 "I"，如下圖。



(5) 當整棵樹畫出來之後，以下列之「後序走訪(postorder traversal)」演算法走訪之，即得次序 FHIGDEBCA。

```

void postorder(Node* p)
{
    if (p!=NULL)
    {
        postorder(p->left);
        postorder(p->right);
        cout << p->data;
    }
}

```

因此，答案為 FHIGDEBCA。

十二、化簡布林函數： $F(A, B, C, D) = \Sigma(2,3,5,6,7,8,10,12,13,14)$ ，結果為？           (12)          。

**解答：**

$F(A, B, C, D) = \Sigma(2,3,5,6,7,8,10,12,13,14)$  的意思是當  $(A, B, C, D)$  的以二進位數值型式排列為  $(0010)_2 = 2$ 、 $(0011)_2 = 3$ 、 $(0101)_2 = 5$ 、 $(0110)_2 = 6$ 、 $(0111)_2 = 7$ 、 $(1000)_2 = 8$ 、 $(1010)_2 = 10$ 、 $(1100)_2 = 12$ 、 $(1101)_2 = 13$ 、 $(1110)_2 = 14$  時，函數  $F$  的值為 1。

於是，函數  $F$  亦可表示為  $F(A, B, C, D) = \overline{A}BC\overline{D} + \overline{A}BCD + \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BCD + A\overline{B}C\overline{D} + A\overline{B}CD + A\overline{B}\overline{C}D + A\overline{B}CD$  將函數  $F$  以卡諾圖(K-map)表示之，如下：

| F                          | AB | $\overline{A}B$ | $\overline{A}\overline{B}$ | $A\overline{B}$ |
|----------------------------|----|-----------------|----------------------------|-----------------|
| CD                         | 0  | 1               | 1                          | 0               |
| $\overline{C}D$            | 1  | 1               | 0                          | 0               |
| $\overline{C}\overline{D}$ | 1  | 0               | 0                          | 1               |
| $C\overline{D}$            | 1  | 1               | 1                          | 1               |

將上圖鄰接項目框選後，可再得下圖：

| F                | AB | $\bar{A}\bar{B}$ | $\bar{A}B$ | $A\bar{B}$ |
|------------------|----|------------------|------------|------------|
| CD               | 0  | 1                | 1          | 0          |
| $\bar{C}D$       | 1  | 1                | 0          | 0          |
| $\bar{C}\bar{D}$ | 1  | 0                | 0          | 1          |
| $C\bar{D}$       | 1  | 1                | 1          | 1          |

因此，函數F可化簡為 $F(A,B,C,D) = \underbrace{A\bar{D}}_{\text{粉色區塊}} + \underbrace{\bar{A}C}_{\text{綠色區塊}} + \underbrace{B\bar{C}D}_{\text{藍色區塊}}$

故答案為  $\bar{A}C + A\bar{D} + B\bar{C}D$ 。

十三、若以f(22)呼叫下列f()函式，總共會印出多少個數字？\_\_\_\_\_ (13) \_\_\_\_\_。

```
void f(int n) {
    printf("%d\n", n);
    while (n != 1) {
        if ((n%2)==1) {
            n=3*n+1;
        }
        else {
            n=n/2;
        }
        printf("%d\n", n);
    }
}
```

**解答:**

以f(22)呼叫f()函式，所輸出的數字序列為22、11、34、17、52、26、13、40、20、10、5、16、8、4、2、1，共有16個。

故答案為16。

十四、G()為遞迴函式，請問G(3,7)執行後回傳值為何？\_\_\_\_\_ (14) \_\_\_\_\_。

```
int G(int a, int x) {
    if (x==0)
        return 1;
    else
        return (a*G(a, x-1));
}
```



解答:

$$\begin{aligned}G(3,7) &= 3 * G(3,6) \\ &= 3 * 3 * G(3,5) \\ &= 3 * 3 * 3 * G(3,4) \\ &= 3 * 3 * 3 * 3 * G(3,3) \\ &= 3 * 3 * 3 * 3 * 3 * G(3,2) \\ &= 3 * 3 * 3 * 3 * 3 * 3 * G(3,1) \\ &= 3 * 3 * 3 * 3 * 3 * 3 * 3 * G(3,0) \\ &= 3 * 3 * 3 * 3 * 3 * 3 * 3 * 1 \\ &= 2187\end{aligned}$$

故答案為 2187。

---

十五、請問下列 C 語言程式，執行完後，最後輸出為何？\_\_\_\_\_ (15) \_\_\_\_\_。

```
int g1=30, g2=20;
int f1(int v) {
    int g1=10;
    return g1+v;
}

int f2(int v) {
    int c=g2;
    v=v+c+g1;
    g1=10;
    c=40;
    return v;
}

int main() {
    g2=0;
    g2=f1(g2);
    printf("%d", f2(f2(g2)));
    return 0;
}
```

解答:

本題中，函數 f1 中的變數 g1 是區域變數；而函數 f2 中的變數 g1 則是全域變數。

當主程式 main() 中的 g2=f1(g2) 被呼叫時，全域變數 g1 的值未被更動，其值為 30，而 f1 的傳回值 g1+v=10+0=10，於是全域變數 g2 的值在 f1 函數呼叫結束後，在 main() 中被改為 10。

當呼叫 f2(f2(g2)) 時，即為呼叫 f2(f2(10))。

f2(10) 執行時，c 的值設定為 10、v 的值 v+c+g1=10+10+30=50、全域變數 g1 的值在 v 計算完畢後被改為 10，而傳回值 v 為 50。

於是， $f_2(f_2(g_2))=f_2(f_2(10))=f_2(50)$ 。

$f_2(50)$ 執行時， $c$  的值仍為 10、 $v$  的值  $v+c+g_1=50+10+10=70$ ，傳回值  $v$  即為 70。

故答案為 70。

---

十六、給定一實數 10110.100011，若按 IEEE 754 標準儲存，以二進位表示結果為\_\_\_\_\_ (16)\_\_\_\_\_。IEEE 754 標準的儲存格式：符號位元：1 個位元，以 0 表示正數；以 1 表示負數。指數部分：8 個位元，以過剩 127(Excess 127)方式表示。尾數部分：23 個位元，從標準化的小數點後開始存起，不足的位元部份補 0。

解答：

IEEE 754 浮點數的格式如下為，

| 符號(1 位元) | 指數(8 位元) | 尾數(23 位元)                    |
|----------|----------|------------------------------|
| X        | XXXXXXXX | XXXXXXXXXXXXXXXXXXXXXXXXXXXX |

二進位數 1011.100011 轉換成 IEEE 754 浮點數儲存格式的步驟如下：

- (1) 先將浮點數轉換成類似科學記號表示法的格式，如：  
 $1.10110.100011_{(2)} = 1.0110100011_{(2)} \times 2^4$
- (2) 10110.100011 為正數，所以符號位元為 0。
- (3) 指數部份則為  $4 + 127 = 131_{(10)} = 10000011_{(2)}$
- (4) 尾數部份為 0.0110100011 的小數部份補 0 到 23 位數，其結果為 01101000110000000000000

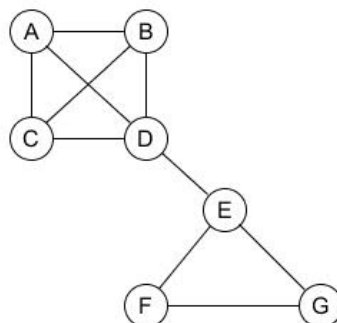
因此，答案為 01000001101101000110000000000000。

---

十七、Ann, Bill, Cherry, David 彼此是好朋友。David 除了認識 Ann, Bill 和 Cherry 外，也還認識 Eva。Eva 除了認識 David，也認識 Frank 和 Grand。Frank 和 Grand 亦是好朋友。若 Ann 想認識 Grand，應該透過誰來介紹呢？\_\_\_\_\_ (17)\_\_\_\_\_。

解答：

下圖為本題好朋友關係的圖(Graph)，



Ann 想要認識 Grand，在上圖中節點 A 與節點 G 之間的路徑上，最直接而關鍵的節點為代表 David 與 Eva 的節點 D 和節點 E。

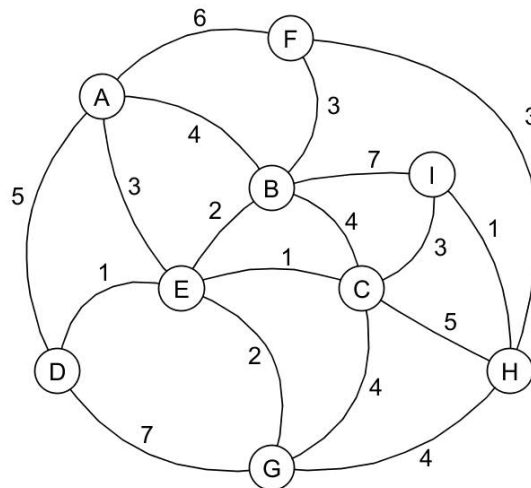
因此，答案為 "David 與 Eva"。

十八、若已知有 9 個城市 {A, B, C, D, E, F, G, H, I} 及連接城市之間的车票費用如下表。請分別計算從城市 A 到達城市 G 與城市 H 所需最低票價 \_\_\_\_\_ (18) \_\_\_\_\_、\_\_\_\_\_ (19) \_\_\_\_\_。

|   | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| A |   | 4 |   | 5 | 3 | 6 |   |   |   |
| B | 4 |   | 4 |   | 2 | 3 |   |   | 7 |
| C |   | 4 |   |   | 1 |   | 4 | 5 | 3 |
| D | 5 |   |   |   | 1 |   | 7 |   |   |
| E | 3 | 2 | 1 | 1 |   |   | 2 |   |   |
| F | 6 | 3 |   |   |   |   |   | 3 |   |
| G |   |   | 4 | 7 | 2 |   |   | 4 |   |
| H |   |   | 5 |   |   | 3 | 4 |   | 1 |
| I |   | 7 | 3 |   |   |   |   | 1 |   |

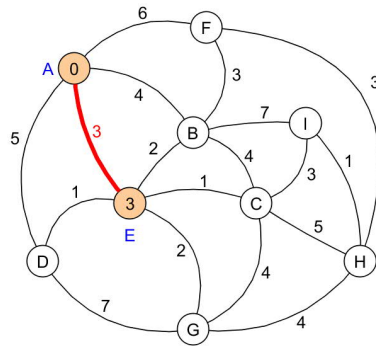
**解答:**

本題表格中，節點 B、C、E 與其它節點的連結最多。若欲將城市之間的關係繪製成圖 (Graph)，宜以節點 B、或節點 C、或節點 E 為起點，如下。

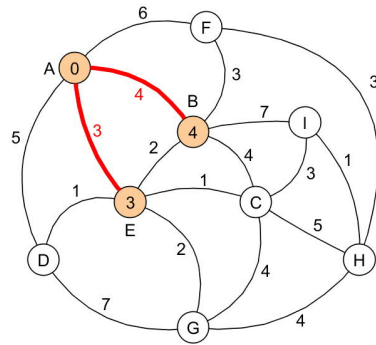


此題為最短路徑問題，以節點 A 為起點，到達各城市的最低票價演算過程如下：

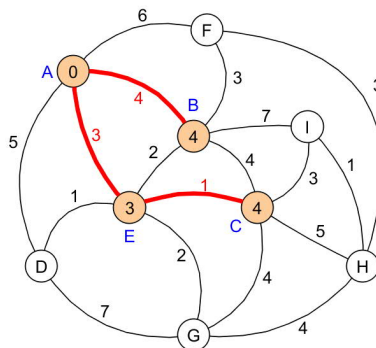
(1) 以 A 為起點，以最低票價可達的城市為 E，票價為 3，如下圖。



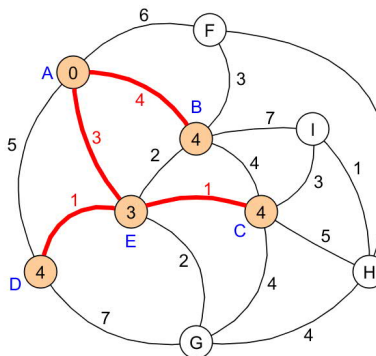
(2) 從{A,E}出發，下一個最低票價可達的城市為B，票價為4，如下圖。



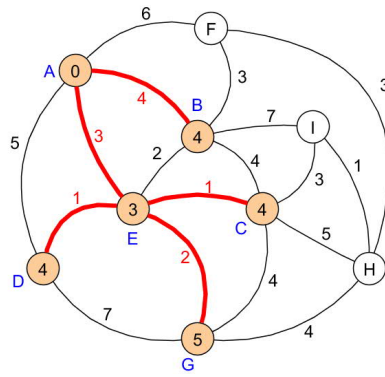
(3) 從{A,B,E}出發，下一個最低票價可達的城市為C，票價亦為4，如下圖。



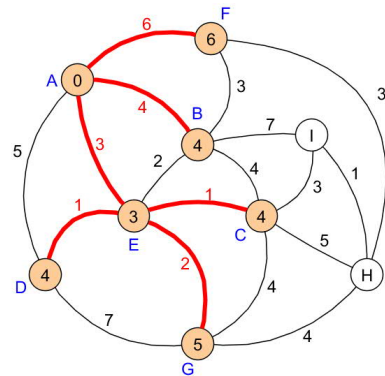
(4) 從{A,B,C,E}出發，下一個最低票價可達的城市為D，票價亦仍為4，如下圖。



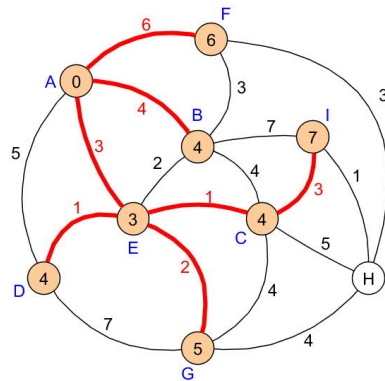
(5) 從{A,B,C,D,E}出發，下一個最低票價可達的城市為G，票價為5，如下圖。



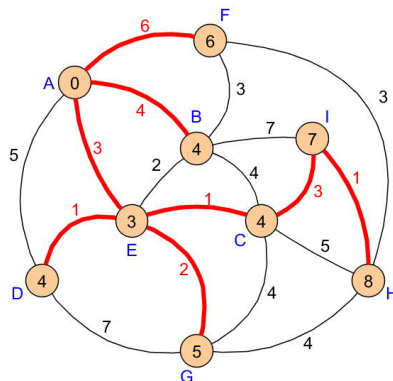
(6) 從{A,B,C,D,E,G}出發，下一個最低票價可達的城市為F，票價為6，如下圖。



(7) 從{A,B,C,D,E,F,G}出發，下一個最低票價可達的城市為I，票價為7，如下圖。



(8) 從{A,B,C,D,E,F,G,I}出發，下一個最低票價可達的城市為H，票價為8，如下圖。



故答案為 5 及 8。

十九、考慮泡沫排序法(bubble sort)、插入排序法(insertion sort)、合併排序法(merge sort)與快速排序法(quick sort)等四種排序法，何者在最佳狀態(best case)與最壞狀態(worst case)下排序  $n$  個元素的陣列時，其時間複雜度均為  $O(n \log n)$  ? \_\_\_\_\_ (20) 。

解答:

| 排序法<br>時間複雜度   | Bubble sort | Insertion sort | Merge sort    | Quick sort    |
|----------------|-------------|----------------|---------------|---------------|
| 最佳(best case)  | $O(n)$      | $O(n)$         | $O(n \log n)$ | $O(n \log n)$ |
| 最壞(worst case) | $O(n^2)$    | $O(n^2)$       | $O(n \log n)$ | $O(n^2)$      |

故答案為 "合併排序法" 。