

105 學年度普通型高級中等學校 資訊學科能力競賽 決賽 筆試試卷

說明：

1. 作答時間 60 分鐘。若須計算或作圖，請利用本試卷的空白處。
2. 本筆試試卷題目為選擇題共 25 題，每題 4 分，滿分為 100 分。
3. 答案必須按題號依序填入「答案卷」上之空格內，否則不予計分。

1. 排序演算法中常以交換的方式進行，交換代表某一個數字和某一個數字互換位置，例如給定一數列 [5 3 2 4 1]，我們想要將此數列從小到大排序，若不做重複的交換的話(同一個數字不能和另一個數字做兩次以上的交換)，則至少要交換兩次

(目標) [1 2 3 4 5]

(初始值) [5 3 2 4 1]

(第一次) [1 3 2 4 5] 交換(1,5)

(第二次) [1 2 3 4 5] 交換(2,3)

若要將數列 [5 7 6 1 4 3 2] 進行排序(小到大)，且不做重複的交換，最少交換幾次後即可完成排序？

- (A) 3
(B) 4
(C) 5
(D) 6
2. 利用較有效率的方法計算下列多項式時，總共需要幾個加、減、乘法的運算？

$$f(x) = 2x^4 + 3x^3 - 3x^2 + 5x - 1$$

- (A) 14
(B) 12
(C) 11
(D) 8
3. 給定下列物品清單，若卡車載重上限為 3500，請問最多可搬移多少價值的物品？

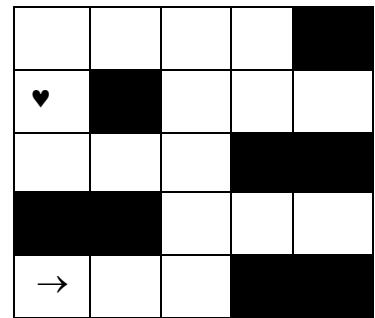
重量	815	906	127	914	633	98	279
價值	547	958	965	158	971	958	486

- (A) 5043
(B) 4885
(C) 4496
(D) 4072

4. 在右圖中，→ 表示機器人起點位置且為目前面對的方向，
 機器人需要走到標註♥的位置，控制機器人的指令有兩個：

- CAN_MOVE(方向)：檢查並回傳是否可往「方向」前進
- MOVE(方向) 表示往「方向」轉後前進一步。

例如 CAN_MOVE(FORWARD) 即回傳是否可往現在面對的方向前進一步；而 MOVE(LEFT) 則會讓機器人左轉並往前推進一步。



右側反覆執行的程式中，方向 A、

方向 B、方向 C 應該分別設成什麼，才
 可以讓機器人以最短的路徑走到♥的位置？

- (A) FORWARD, LEFT, RIGHT
 (B) RIGHT, FORWARD, LEFT
 (C) LEFT, FORWARD, RIGHT
 (D) LEFT, RIGHT, FORWARD

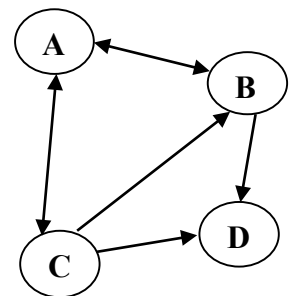
```
REPEAT UNTIL (Goal(♥))
{
    IF CAN_MOVE( 方向 A )
        MOVE( 方向 A )
    IF CAN_MOVE( 方向 B )
        MOVE( 方向 B )
    IF CAN_MOVE( 方向 C )
        MOVE( 方向 C )
}
```

5. 小城镇中的電車路線圖如下，其中 A, B, C 和 D 表示四個站，單一箭頭的連接線表示電車從端點為起站開向有箭頭端的終站；雙箭頭的連接線表示兩端點的站之間雙向都有電車互通。

若以一個二維陣列 Map[i][j] 來記錄這個電車路線圖，

Map[i][j]=1 表示從站 i 到站 j 有一條電車路線，否則 Map[i][j]=0。

請問下列哪幾個陣列可以用以表示右側電車路線圖？



	0	0	0	0
[1]	1	0	1	0
	0	1	0	1
	1	1	1	0

	0	1	0	0
[2]	1	0	1	1
	0	1	0	0
	1	0	1	0

	0	1	1	1
[3]	1	0	0	1
	0	0	0	0
	0	1	1	0

	0	0	0	0
[4]	1	0	1	1
	0	1	0	1
	1	1	0	0

- (A) 只有陣列 [1] 是可能的
 (B) 只有陣列 [1] 跟 [3] 是可能的
 (C) 只有陣列 [1] 跟 [4] 是可能的
 (D) 只有陣列 [2] 跟 [3] 是可能的

6. e^x 的近似解可以用下列式子表示，

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \cdots + \frac{x^{10}}{10!}$$

右方程式碼“(a)”處該填入哪個計算式才能正確計算 e^x ？

- (A) $\text{sum} + x^n / t$
- (B) $\text{sum} + x^t / n$
- (C) $\text{sum} + x * t / n$
- (D) $\text{sum} + x / t * n$

```
scanf("%d", &x);
sum=1;
n=0;
t=1;
while (n<10) {
    n=n+1;
    t=t*n;
    sum = (a);
}
printf("e^x = %d", sum);
```

7. 右側程式擬將陣列 **array** 的元素進行升冪排序，請問空白處要填入哪個計算式才能正確排序？

- (A) n
- (B) $n - i$
- (C) $n - i - 1$
- (D) $n - 1$

```
// array 長度為 n
for (int i=0; i < n-1; i++){
    for (int j=0; j < (a); j++){
        if (array[j+1] < array[j]){
            int temp = array[j];
            array[j] = array[j+1];
            array[j+1] = temp;
        }
    }
}
```

8. 給定右側函式，**f(10)** 回傳值為何？

- (A) 55
- (B) 57
- (C) 105
- (D) 193

```
int f(int n) {
    if (n<=3) return 1;
    return f(n-1)+f(n-2)+f(n-3);
}
```

9. 右測程式片段執行後輸出為何？

- (A) 4
- (B) 10
- (C) 16
- (D) 256

```
x = 1;
y = 10;
for (int i=1; i<=2; i++)
    x = x + x;
for (int i=1; i<=2; i++)
    x = x * x;
y = x;
x = y;
printf ("%d\n", x);
```

10. 右測程式片段執行後輸出為何？

- (A) 0
- (B) 5
- (C) 10
- (D) 15

```
int x=0, y=10; // 全域變數

int swap(int x, int y) {
    int temp = x;
    x = y;
    y = temp;
    return 0;
}

int main() {
    int x=5, ans=0;
    ans = ans + x;
    swap(x, y);
    ans = ans + y;
    printf("%d\n", ans);
    return 0;
}
```

11. 給定 adjust() 函式及陣列 a[] 的內容如下：

```
void adjust (int a[], int root, int n) {
    int child, rootkey;
    int temp;
    temp = a[root];
    rootkey = a[root];
    child = 2 * root;
    while (child <= n) {
        if ((child < n) && (a[child] < a[child+1]))
            child = child + 1;
        if (rootkey > a[child])
            break;
        else {
            a[child/2] = a[child];
            child = child * 2;
        }
    }
    a[child/2] = temp;
}
```

i	1	2	3	4	5	6	7	8	9	10
a[i]	26	5	77	1	61	11	59	15	48	19

若 n=10，則執行以下 for 迴圈之後，a[5] 的內容為？

```
for (i=n/2; i>0; i=i-1) adjust(a, i, n);
```

- (A) 11 (B) 19 (C) 26 (D) 61

12. 給定 nfind() 函式如下：

```
int nfind(char *string, char *pat) {
    int i, j, start = 0;
    int lasts = strlen(string) - 1;
    int lastp = strlen(pat) - 1;
    int endmatch = lastp;
    for (i = 0; endmatch <= lasts; endmatch++, start++) {
        if (string[endmatch] == pat[lastp])
            for (j=0, i=start; j<lastp && string[i]==pat[j];
                i++, j++);
        if (j == lastp)
            return start; /* successful */
    }
    return -1;
}
```

若 pat="aab"、string="ababbaabaaabb"，nfind(string, pat) 回傳值為何？

- (A) 5 (B) 7 (C) 9 (D) -1

13. 右側程式片段執行後會輸出多少個“#”？

- (A) 320
(B) 300
(C) 285
(D) 280

```
int i = 20;
while (i > 0) {
    for (int j=i; j<=i+15; j=j+1)
        printf("#");
    i = i-1;
}
```

14. 背包客找尋旅館時會以離捷運站距離及價格為考量，若旅館 t 比旅館 t' 近捷運站，則在距離條件上旅館 t 優於旅館 t' 。若旅館 t 比旅館 t' 價格便宜，則在價格條件上旅館 t 優於旅館 t' 。若旅館 t 的兩個條件都優於旅館 t' ，或是一個條件相同另一個條件較優，都可說在整體條件上旅館 t 優於 t' 。候選旅館則是沒有其他旅館整體條件比它優的旅館。右側虛擬碼程式可挑出候選旅館，當旅館 t 整體條件優於旅館 t' ，Compare(t, t') 將回傳 1；若旅館 t' 整體條件優於旅館 t ，則回傳 2；否則回傳 0。

若資料庫中共有 4 筆旅館資料：

$H = \{(200m, 2000 \text{ 元}), (120m, 3000 \text{ 元}), (500m, 2500 \text{ 元}), (150m, 3200 \text{ 元})\}$ ，第一個數字代表離捷運站距離，第二個數字為價格。若要找出所有候選旅館，Compare(t, t') 共會被執行幾次？

- (A) 4 次 (B) 5 次 (C) 6 次 (D) 8 次

```
C = ∅ ;
for each hotel t in H do {
    check ← True;
    for each hotel t' in C do {
        r ← Compare(t, t');
        if (r==2) {
            check = False;
            break;
        }
        if (r==1) {
            C ← 將 t' 從 C 中移除;
        }
    }
    if (check) {
        C ← 將 t 加入 C;
    }
}
return C;
```

15. (續上題) 若現有 20 間旅館 (以集合 H 表示), 但不知道這些旅館的順序, 但已知最後找出的候選旅館共有 4 間 (以集合 C 表示), 請問上述虛擬碼程式中 $\text{Compare}(t, t')$ 最多會執行幾次?
- (A) 64 次
(B) 70 次
(C) 80 次
(D) 190 次
16. 一個有 105 個數值的遞增數列, 利用二元搜尋法進行搜尋, 最多需搜尋 m 次, 最少需搜尋 n 次, 便可確定要搜尋的數值是否在數列中, 請問 $m+n$ 為多少?
- (A) 6
(B) 7
(C) 8
(D) 9
17. 有 X 、 Y 、 Z 三個數, 分別是 $X=(E4)_{16}$ 、 $Y=(343.56)_8$ 、 $Z=(11100100.11)_2$, 請問這三個數的大小關係為何?
- (A) $X > Y > Z$
(B) $Y > X > Z$
(C) $Z > Y > X$
(D) $Z > X > Y$
18. 某雜湊表(hash table)有 11 個空格, 編號為 0 到 10。假設雜湊函數 (hash function) 為 $h(k) = k \bmod 11$, 且此雜湊表使用平方探測法 (quadratic probing): $h(k, i) = \{[h(k) + i^2] \bmod 11, i=1, 2, 3, \dots\}$ 處理碰撞 (collision)。依此方法, 若將 27、35、21、4、59、75、33、41 等 8 個數字依序存入後, 則此時編號 3 的空格所存之數字為何?
- (A) 75
(B) 33
(C) 41
(D) 沒有數字
19. 一個二元樹使用後序遍歷 (Postorder Traversal) 結果為 0, 6, 3, 7, 1, 4, 2, 5, 8, 9, 使用中序遍歷 (Inorder Traversal) 結果為 0, 4, 1, 6, 7, 3, 2, 8, 5, 9, 請問對此二元樹使用前序遍歷 (Preorder Traversal) 結果會為何?
- (A) 9, 8, 2, 5, 4, 0, 1, 7, 6, 3
(B) 6, 3, 7, 0, 1, 4, 2, 5, 8, 9
(C) 9, 8, 2, 4, 0, 1, 7, 6, 3, 5
(D) 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

20. 假設系統中只有四個程序 P1, P2, P3 與 P4 欲執行, 且每個程序分別需要花費 6ms、8ms、7ms 與 3ms 的 CPU 時間 (CPU time)。若採用最短時間優先(shortest-job-first) 排程法時, 在不考慮各種額外花費的時間(overhead)下, 則這四個程序的平均等待時間 (不包含程序本身運行時間) 為多少?

- (A) 10ms
- (B) 11ms
- (C) 6ms
- (D) 7ms

21. 已知若問題 A 存在解答, 則問題 B 也存在解答。下列兩推斷是否正確?

Q1: 若問題 A 不可解, 問題 B 也不可解;

Q2: 若問題 B 不可解, 問題 A 也不可解。

- (A) Q1 正確、Q2 正確
- (B) Q1 正確、Q2 不正確
- (C) Q1 不正確、Q2 正確
- (D) Q1 不正確、Q2 不正確

22. 右圖是下列那一個邏輯運算式的真值表?

- (A) (A OR B) AND (B OR C)
- (B) (A AND C) OR (B AND C)
- (C) (A AND B) OR (B AND C)
- (D) (A OR C) AND (B OR C)

A	B	C	結果
F	F	T	F
F	T	F	F
F	T	T	T
T	T	F	F

23. 合併排序法(merge sort)的設計最主要是採用哪一種演算法設計的概念?

- (A) divide and conquer (分治法)
- (B) greedy algorithm (貪婪演算法)
- (C) branch-and-bound (修剪法)
- (D) backtracking(回溯法)

24. 已知有一個二元樹, 以前序遍歷(Preorder Traversal)走訪節點的順序為 C, D, A, B, E, F, G, H。以中序遍歷(Inorder Traversal)走訪節點的順序為 A, D, B, C, G, F, H, E。以下敘述何者錯誤?

- (A) C 為根節點
- (B) F 是 G 的父節點
- (C) E 為一個葉節點
- (D) A 為 B 的兄弟節點

25. 給定以下指令

- PUSH 用來將一筆資料放入一個堆疊(stack)中，
- POP 用來從目前堆疊中的內容取出一筆資料，並將取出資料存入一個最大值優先佇列(Priority queue)，
- DEMAXQUEUE 用來從最大值優先佇列中取出並輸出其中的最大值資料

若執行右側指令後所得到的輸出資料依序為 3, 2, 1。請問
資料 1、資料 2、資料 3 應分別為哪個數字？

- (A) 資料 1 為 3, 資料 2 為 2, 資料 3 為 1
(B) 資料 1 為 2, 資料 2 為 3, 資料 3 為 1
(C) 資料 1 為 1, 資料 2 為 3, 資料 3 為 2
(D) 以上皆可能

```
PUSH (資料 1)
PUSH (資料 2)
POP
POP
DEMAXQUEUE
PUSH (資料 3)
POP
DEMAXQUEUE
DEMAXQUEUE
```