

# 114年特種考試地方政府公務人員及 離島地區公務人員考試試題

考試別：地方政府公務人員考試

等別：四等考試

類科：資訊處理

科目：程式設計概要

考試時間：1小時30分

座號：\_\_\_\_\_

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

(三)本科目除專門名詞或數理公式外，應使用本國文字作答。

一、給定  $n \times m$  ( $n$  行  $m$  列) 影像的各像素灰度值，灰度值為 0 至 255 之間的整數 (包含 0 和 255)，像素的相鄰性是指當前像素與周邊像素的鄰接性質，八鄰域 (eight neighbors) 包括對應當前像素位置的上、下、左、右、左上、右上、左下、右下像素位置，換言之，為原像素及其上下左右與對角線相鄰八個像素位置，如下圖所示：

2	1	8
3	P	7
4	5	6

當對此影像進行降低雜訊的模糊化處理時：1. 影像四周最外側的像素灰度值不變；2. 影像中間各像素新灰度值由原像素及其八鄰域像素原灰度值的平均值 (捨入到最接近的整數) 取代原像素值。輸入資料第一行包含兩個整數  $n$  和  $m$ ，表示影像中像素點的行數和列數， $1 \leq n \leq 100$ ， $1 \leq m \leq 100$ ，接下來  $n$  行，每行  $m$  個整數，表示影像中每個像素灰度值。相鄰兩個整數之間用單個空格隔開，每個元素均在 0~255 之間。例如一個  $5 \times 6$  影像及其各像素灰度值如下所示：

5 6  
100 0 120 0 50 70  
50 110 200 0 0 180  
70 50 100 120 200 0  
150 120 50 50 100 10  
10 110 0 50 120 30

請使用 Java 或 C++ 實作一完整程式，讀入一個影像行數、列數和像素灰度值，並使用八鄰域像素灰度值對影像進行降低雜訊的模糊化處理，再輸出模糊化後影像。(20 分)

二、請問以  $a(7,20)$  呼叫下列  $a()$  函式，函式執行完後其回傳值為何？並需詳述執行過程。(20 分)

```
int a(int n, int m) {  
    if (n >= 10) {  
        if (m < 16) {  
            return n + m;  
        }  
        else {  
            return a(n, m-2) + m;  
        }  
    }  
    else {  
        return a(n+1, m) + n;  
    }  
}
```

三、若以  $f(30)$  呼叫下列  $f()$  函式，會列印出那些數字？(20 分)

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

四、給定下列程式，其中 t 被宣告為全域變數，請問程式執行後輸出為何？  
(20 分)

```
int t=10; //全域變數

void sum(int a) {
    int t=20;
    for( ; a>=0; a=a-5) {
        printf("%d,", t);
        t=t+a;
        printf("%d,", t);
    }
}
int main () {
    printf("%d,", t);
    sum(t);
    printf("%d,", t);
    t=15;
    printf("%d", t);
    return 0; }
```

五、單鏈結串列 (Singly Linked List) 特點是鏈結串列的連結方向為單向，對單鏈結串列的訪問要通過從頭部開始，依序往下讀取，每個節點 (node) 包含兩部分：存儲數據的資料 (data) 和指向下一個節點的指標 (pointer)，節點定義如下：

```
struct Node{
    int data;
    struct Node *next;
}
```

單鏈結串列中的所有節點通過指標串聯，最後一個節點的指標指向空值 (null)，表示單鏈結串列結尾。請使用 Java 或 C++ 實作一完整程式，通過從頭部 (head) 開始，依序往下讀取，訪問一單鏈結串列操作，輸出單鏈結串列中節點個數，若單鏈結串列中不存在任何節點，為一個空鏈結串列，則輸出-1。(20 分)