

主題 5.4



題型5-18 遞迴及非遞迴程式

【重點說明】

在一般的遞迴演算法中，常將原來的問題先分割成幾個較小的問題，且小問題性質與原來的大問題一樣，然後將這些小問題一一解決，合併後就可得到較大問題的解答，這種解決問題的方法稱之為個個擊破法（divide and conquer）。遞迴演算法是個個擊破法的一種表現方式。遞迴（recursive）程式與非遞迴（nonrecursive）程式最主要的差別，在於以遞迴方式設計程式較簡易且容易了解。但遞迴方式也是有它的缺點，最大的缺點就在於，以遞迴方式執行時，會占用較多記憶體空間和花費較多的CPU時間。下表為遞迴與非遞迴優缺點的比較表。

	優點	缺點
遞迴	程式簡潔明確且節省記憶體空間。	參數的堆疊存取較費時。
非遞迴	節省執行時間。	程式較長，浪費記憶體空間。

註：非遞迴（nonrecursive）的版本（version）通常被稱做iterative version。



範題

《常考題》

任選一程式語言，寫出求二正整數的最大公因數GCD的程式。

【解】

GCD非遞迴版本：

```
int gcd(int m, int n)
{
    if (m<0)
        m=-m; //最大公因數為正
    if (n<0)
        n=-n; //最大公因數為正
    while (n!=0) //利用 $d|a$ 且 $d|b \Rightarrow d|ma \pm nb$ ，其中  $m, n \in Z$ 。
    {
        int temp = m;
        while (temp>=n)
            temp = temp - n;
        m = n; //let m always bigger than n
        n = temp;
    }
    return m;
}
```

GCD遞迴版本：

```
int gcd(int x, int y) {
    if (x mod y) // x mod y !=0
        return gcd(y, x mod y)
    else // if x mod y ==0
        return y;
}
```



範題

《重要題》

有一Fibonacci 數學函式如下：

$$F(n) = F(n-1) + F(n-2), n \geq 2。$$

$$F(1)=1, F(0)=0。$$

請分別運用：(1)Recursive；(2)Non-recursive兩種方法，寫出上列函數的完整程式，用C/C++。

【解】

(1)遞迴版本：

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

(2)非遞迴版本：

```
typedef int result_value, input_value; //使用者自定型別
result_value Fibonacci(input_value n) {
    if(n<=1) return n ;
    int Fibonacci_n0=0 ; Fibonacci_n1=1 ;
    for(int i=2 ; i<=n ; i=i+1){
        Fibonacci_now=Fibonacci_n0+Fibonacci_n1 ;
        Fibonacci_n0=Fibonacci_n1 ;
        Fibonacci_n1=Fibonacci_now ;
    }
    return Fibonacci_now ;
}
```



範題

《基本題》

請分別運用(1)Recursive；(2)Non-recursive兩種方法，寫出下列函數的完整程式，求等差級數和，計算 $f(n) = 1 + 2 + 3 \cdots + n = ?$ （使用C/C++語言）

【解】

(1)使用Loop迴圈，即非遞迴方式：

```
typedef int result_value,input_value; //使用者自定型別
result_value SUM_LOOP(input_value n) {
    int Sum = 0;
    For (int i=1 ,i<= n,i++)
        Sum=Sum+i ;
    return Sum;
}
```

(2)使用遞迴方式：

```
typedef int result_value,input_value; //使用者自定型別
result_value SUM_RECURSION ( input_value n ) {
    int Sum = 0;
    If (n<=1)
        return Sum;
    else
        return Sum = n + SUM_RECURSION( n-1);
}
```



範題

請用C/C++/Java任一語言寫一個function，用以傳回第n項的費式級數（Fibonacci），如下說明：

```
int Fib (int n); //第n項的費式級數
```

Fib (1) = 1, Fib (2) = 1, Fib (3) = Fib (1) + Fib (2) = 2, …,