

第四章

結構化程式設計的流程控制



4-6 迴圈結構(Loop)



【引言】

在我們的日常生活中，有許多事都是具有重覆性的，例如一天有 24 小時，一星期有七天，同一門課要上 18 次才能拿到學分，一個大學學位可以要修 132 個學分才能拿到。或每一位學生每一次月考要考五科，一學期要考三次月考，諸如此類的問題，如果只靠人工處理必定是一件非常煩的工作。

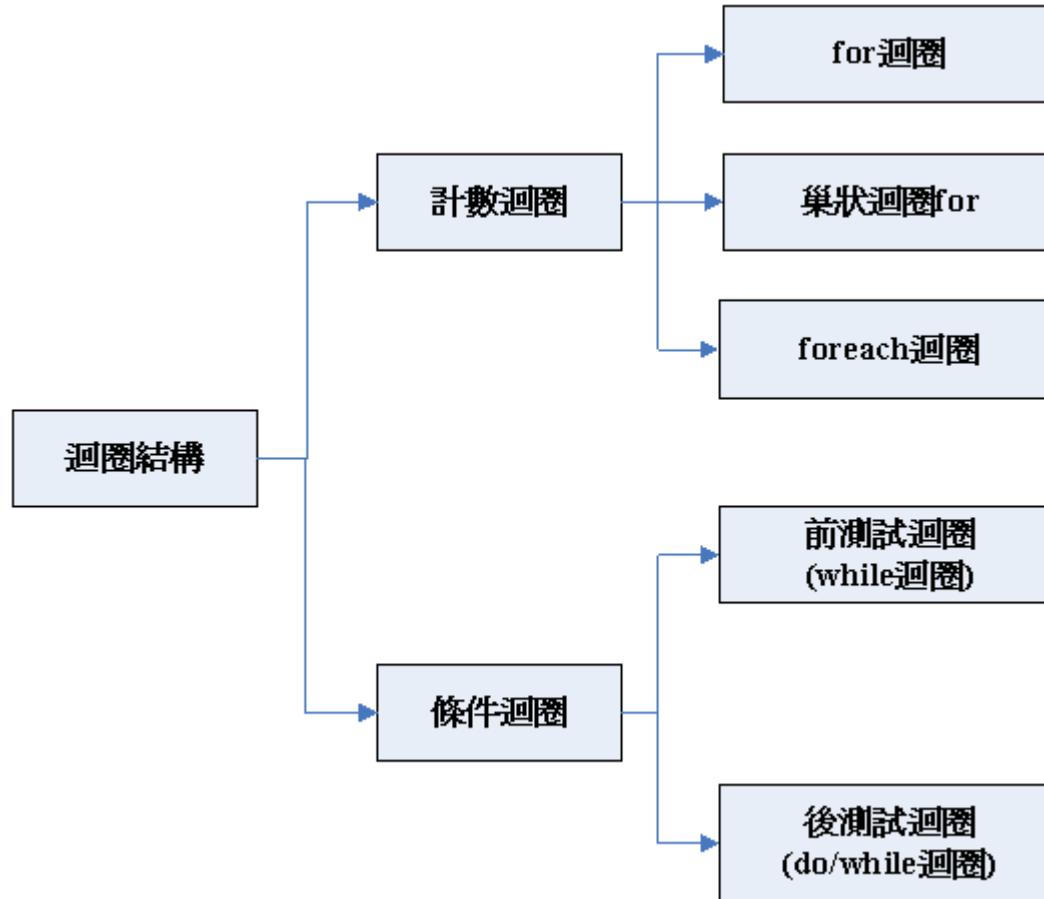
【示意圖】

一天有 24 小時	一星期有七天
	

【定義】是指讓某一段敘述反覆執行多次的程式。

【使用時機】一再重複性的動作。

【分類架構圖】



【範例】

一、機器人繞一個正方形(可使用for迴圈)

二、機器人反覆某一動作(可使用while與do/while迴圈)

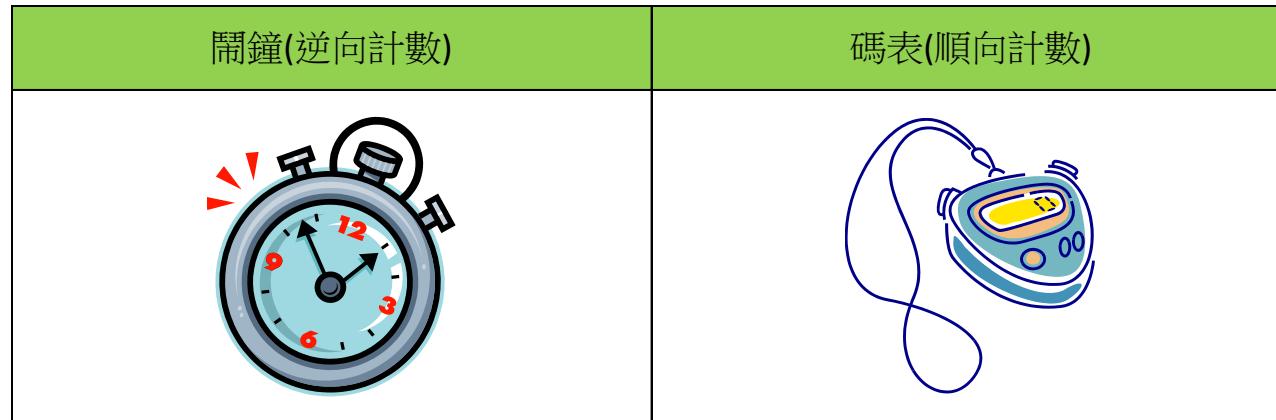
【分類】「計數迴圈」與「條件迴圈」



4-6-1 計數迴圈

【定義】是指依照「計數器」的設定值，來依序重複執行。

【示意圖】鬧鐘與碼表



【使用時機】已知程式的執行次數固定且重覆時，使用此種迴圈最適合。

【種類】

1. 單層for迴圈
2. 巢狀for迴圈
3. foreach迴圈

4-6-1.1 單層for迴圈



【定義】是單層次的迴圈結構。

【使用時機】適用於「單一變數」的重覆變化。

【典型例子】 $1+2+3+\dots+10$

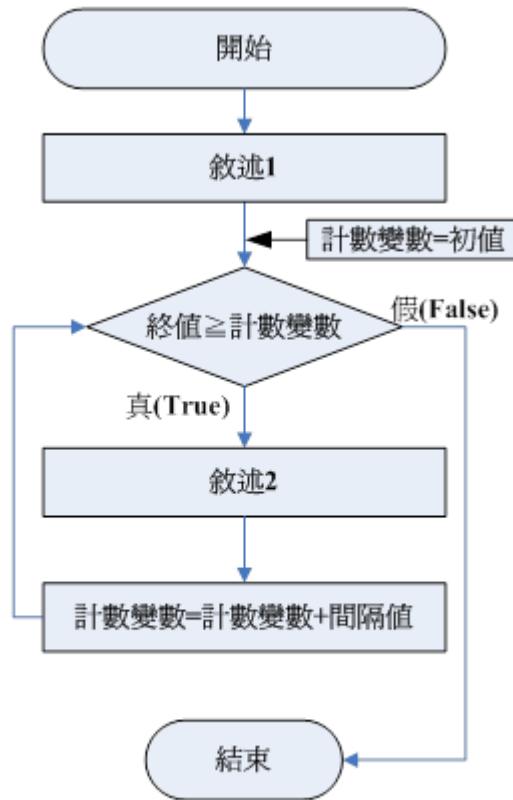
【語法】

```
for (計數變數=初值;終值條件;間隔值)
{
    程式區塊 1;
    [break;]
    [continue;]
    程式區塊 2;
}
```

【動作原理】

- 一、「計數變數」設定為「初值」，然後執行迴圈內之「程式區塊1」。
- 二、直到執行「程式區塊2」完成後，「計數變數」會自動加上「間隔值」。
- 三、此時，迴圈結構會自動檢查「計數變數」是否大於「終值」，
 - (一)若超過則跳出迴圈；
 - (二)否則將繼續執行迴圈敘述；
 - (三)直到「計數變數」大於終值為止。

【流程圖】



【實例】1加到10的三種常見的求法

過程 求法	迴圈敘述	I 值的變化過程	迴圈內的 執行次數	總和
累加	for (i=1;i<=10;i++)	1,2 , 3...9 , 10	10	55
奇數和	for (i = 1; i <= 10; i+=2)	1,3,5,7,9	5	25
偶數和	for (i = 2; i <= 10; i+= 2)	2,4,6,8,10	5	30



【延伸學習】

(1) for敘述可用來重覆執行迴圈內的敘述。

【舉例】`for (i = 1; i <= 20; i += 3)`

其重覆次數的公式為：

①重覆次數 = (終值 - 初值) \ 間隔值 + 1

重覆次數 = $(20-1) \backslash 3 + 1 = 19 \backslash 3 + 1 = 6 + 1 = 7$ (註：「\代表取商數」)

②真正的終值 = 初值 + (次數-1)*間隔值 = $1 + (7-1) * 3 = 19$

③總和(利用數學上的梯形面積計算方法) = $\frac{\text{上底} + \text{下底}}{2} * \text{高}$

總和 = $\frac{\text{終值} + \text{初值}}{2} * \text{次數} = \frac{19+1}{2} * 7 = 70$

(2) 動作模式：首先計數變數設定為初值，然後執行迴圈內之敘述，「計數變數」將加上"間隔值"，並檢查「計數變數」是否大於"終值條件"，若超過則跳出迴圈；否則將繼續執行迴圈敘述，直到「計數變數」大於終值條件的終值為止。

(3) for敘述的間隔值必須和初值、終值條件的終值配合：

- ① 當初值 > 終值條件的終值時，間隔值應為負值，例如：10(初值)>1(終值)，間隔值為 -2(即 $i-= 2$)，其執行過程為：10、8、6、4、2。
- ② 當初值 < 終值條件的終值時，間隔值應為正值，例如：1(初值)<10(終值)，間隔值為2(即 $i+= 2$)，其執行過程為：1、3、5、7、9。

【實作1】請利用for迴圈來撰寫1加到10的程式。

行號	程式檔名：ch4-6-1-1A.java
01	<code>public class ch4_6_1_1A {</code>
02	<code> public static void main(String[] args) {</code>
03	<code> int Sum=0;</code>
04	<code> for (int i = 1; i <= 10; i++)</code>
05	<code> Sum = Sum + i;</code>
06	<code> System.out.println("Sum="+ Sum);</code>
07	<code> }</code>
08	<code>}</code>

【說明】

行號03：宣告及設定初值

行號04：利用for迴圈來控制10次

行號05：Sum變數值每一次加i變數值，亦即進行「累加」功能

行號06：將Sum變數值(亦即總和)顯示在螢幕上。

【實例2】迴文判斷

【說明】請由表單中讀入一個欲判斷的數字，若此數字為迴文(左右讀起均同，例如12321)、則印出此數字及“是迴文”，若不是則印出此數字及“不是迴文”

【輸入範例】12321

【輸出報表】12321是迴文

行號	程式檔名：ch4-6-1-1B.java
01	<code>import java.util.Scanner; //載入Scanner類別套件</code>
02	<code>public class ch4_6_1_1B</code>
03	<code>{</code>
04	<code> public static void main(String[] args)</code>
05	<code> {</code>
06	<code> //宣告及初值設定</code>
07	<code> int D1, D2=0, D3, i;</code>
08	<code> int[] A=new int[10];</code>
09	<code> String result;</code>
10	<code> //輸入</code>
11	<code> Scanner inNumber=new Scanner(System.in);</code>
12	<code> System.out.print("請輸入一串數字：");</code>
13	<code> String strNumber=inNumber.next();</code>
14	<code> //處理</code>
15	<code> D1=Integer.parseInt(strNumber);</code>
16	<code> D3 = D1;真理大學 資訊工程學系</code>

```

17     for(i=1;i<=10;i++)
18     {
19         A[i] = D1 % 10;
20         D1 = (int)(D1 / 10);
21         D2 = D2 * 10 + A[i];
22         if (D1 == 0)
23             break;
24         }
25         if (D2 == D3)
26             result="是迴文";
27         else
28             result="不是迴文";
29         //輸出
30         System.out.println()//換行
31         System.out.println(D3 + result);
32     }
33 }
```

【執行過程】

I	A [i]	D1 = D1 \10	D2 = D2 *10 + A [i]
1	A [1]=1	1232	0*10+1=1
2	A [2]=2	123	1*10+2=12
3	A [3]=3	12	12*10+3=123
4	A [4]=2	1	123*10+2=1232
5	A [5]=1	0	1232*10+1=12321

【註】本題中的「陣列」真題大串流資訊工程請參閱第五章。

4-6-1.2 巢狀for迴圈



【定義】是多層次的迴圈結構。

【使用時機】適用於「兩個或兩個以上變數」的重覆變化。

【典型例子】九九乘法表

【語法】

```
for (計數變數 1=初值 1;終值條件 1;間隔值)
{
    for (計數變數 2=初值 2;終值條件 2;間隔值)
    {
        程式區塊 1;
        [break;]
        [continue;]
        程式區塊 2;
    }
}
```

外
迴
圈

內
迴
圈

【說明】

當外迴圈的計數變數1值增加一次時，則內迴圈必須Run一回合，也就是說，計數變數2必須從「初值」執行到「終值條件」結束。基本上，常見的巢狀迴圈有下列三種不同的變化：

第一種：基本型巢狀	第二種：擴張型巢狀	第三種：多層型巢狀
<pre>for(...;...;...) for(...;...;...)</pre>	<pre>for(...;...;...) for(...;...;...) for(...;...;...)</pre>	<pre>for(...;...;...) for(...;...;...) for(...;...;...) for(...;...;...)</pre>

【實例】九九乘法表



```
int i, j;
for (i = 1; i <= 9; i++)
{
    for (j = 1; j <= 9; j++)
    {
        //程式區塊
    }
}
```

【解析】

上述的九九乘法表共有內外兩層迴圈，當外迴圈「*i*變數」變動1次時，則內迴圈則要執行9次，所以，當外迴圈「*i*變數」由1~9時，內迴圈要執行81次。其執行過程如下所示：

外迴圈 <i>i</i>	內迴圈 <i>j</i>								
	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	1	2	3	4	5	6	7	8	9
3	1	2	3	4	5	6	7	8	9
4	1	2	3	4	5	6	7	8	9
5	1	2	3	4	5	6	7	8	9
6	1	2	3	4	5	6	7	8	9
7	1	2	3	4	5	6	7	8	9
8	1	2	3	4	5	6	7	8	9
9	1	2	真理大學	資訊工程學系	6	7	8	9	

【實例1】：九九乘法表程式

程式檔案名稱 ch4-6-1-2A.java

```
01 public class ch4_6_1_2A
02 {
03     public static void main(String[] args)
04     {
05         int i,j;
06         for(i=1;i<=9;i++)          //外迴圈
07         {
08             for(j=1;j<=9;j++)      //內迴圈
09             {
10                 System.out.print(i + "*" + j + "=");
11                 System.out.print(i*j + "\t");
12             }
13             System.out.println();   //換行
14         }
15     }
16 }
```

【程式說明】

當外迴圈*i*=1時，則內迴圈*j*要執行1~9，就會產生如下的情況：

$1*1=1$ $1*2=2$ $1*3=3$ $1*4=4$ $1*5=5$ $1*6=6$ $1*7=7$ $1*8=8$ $1*9=9$

直到*i*到9，就可以產生九九乘法表。

當 *i*=1 時，則 *j* 從 1 To 9

	$1*1=1$	$1*2=2$	$1*3=3$	$1*4=4$	$1*5=5$	$1*6=6$	$1*7=7$	$1*8=8$	$1*9=9$
$2*1=2$	$2*2=4$	$2*3=6$	$2*4=8$	$2*5=10$	$2*6=12$	$2*7=14$	$2*8=16$	$2*9=18$	
$3*1=3$	$3*2=6$	$3*3=9$	$3*4=12$	$3*5=15$	$3*6=18$	$3*7=21$	$3*8=24$	$3*9=27$	
$4*1=4$	$4*2=8$	$4*3=12$	$4*4=16$	$4*5=20$	$4*6=24$	$4*7=28$	$4*8=32$	$4*9=36$	
$5*1=5$	$5*2=10$	$5*3=15$	$5*4=20$	$5*5=25$	$5*6=30$	$5*7=35$	$5*8=40$	$5*9=45$	
$6*1=6$	$6*2=12$	$6*3=18$	$6*4=24$	$6*5=30$	$6*6=36$	$6*7=42$	$6*8=48$	$6*9=54$	
$7*1=7$	$7*2=14$	$7*3=21$	$7*4=28$	$7*5=35$	$7*6=42$	$7*7=49$	$7*8=56$	$7*9=63$	
$8*1=8$	$8*2=16$	$8*3=24$	$8*4=32$	$8*5=40$	$8*6=48$	$8*7=56$	$8*8=64$	$8*9=72$	
$9*1=9$	$9*2=18$	$9*3=27$	$9*4=36$	$9*5=45$	$9*6=54$	$9*7=63$	$9*8=72$	$9*9=81$	

4-6-2 條件迴圈(Do/Loop)



【引言】

一般而言，如果我們不能預先知道迴圈的次數的話，則必須要使用「條件迴圈」來解決。

【例如】

王爸爸第一天給小明10元，第二天給20元，第三天給30元，請問小明要儲存到5000元需要多少天？諸如此類的題目，最好使用條件式迴圈來處理比較容易。

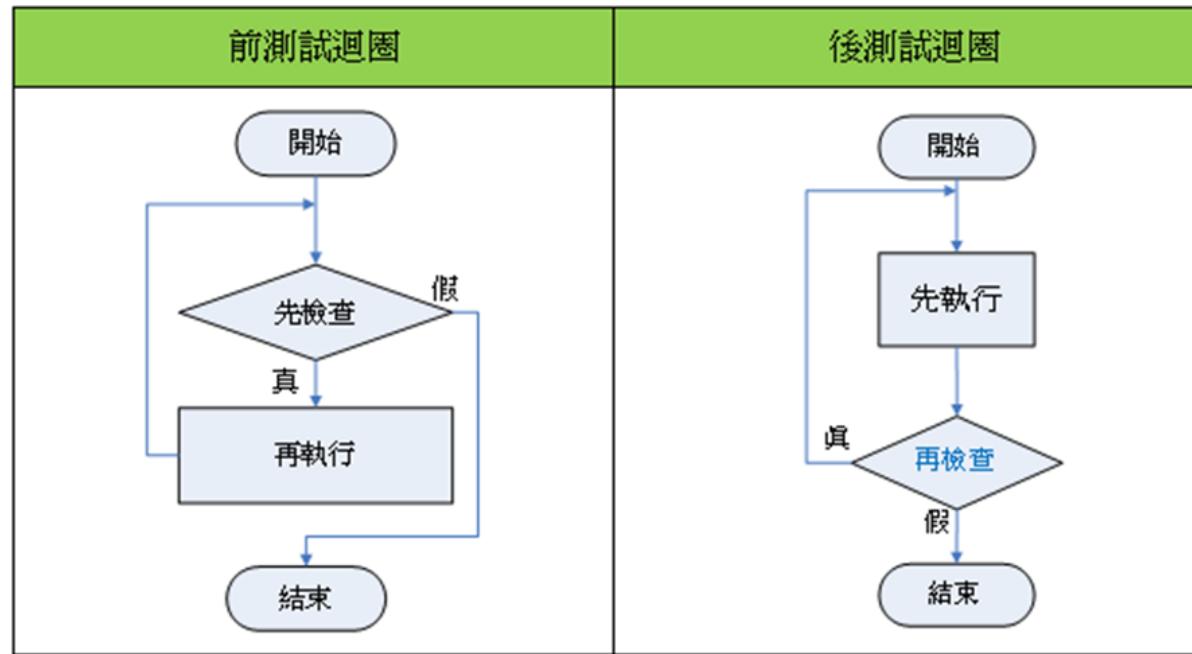
【示意圖】

情境	小明要儲存到5000元需要多少天？
 王爸爸 真理大學 小明 資訊工程學系	

【分類】

1. 前測試迴圈(先判斷條件式，再執行迴圈)。如：while
2. 後測試迴圈(先執行迴圈，再判斷條件式)。如：do/while

【圖解說明】



4-6-2-1 前測試迴圈



【定義】先判斷條件式，再執行迴圈。

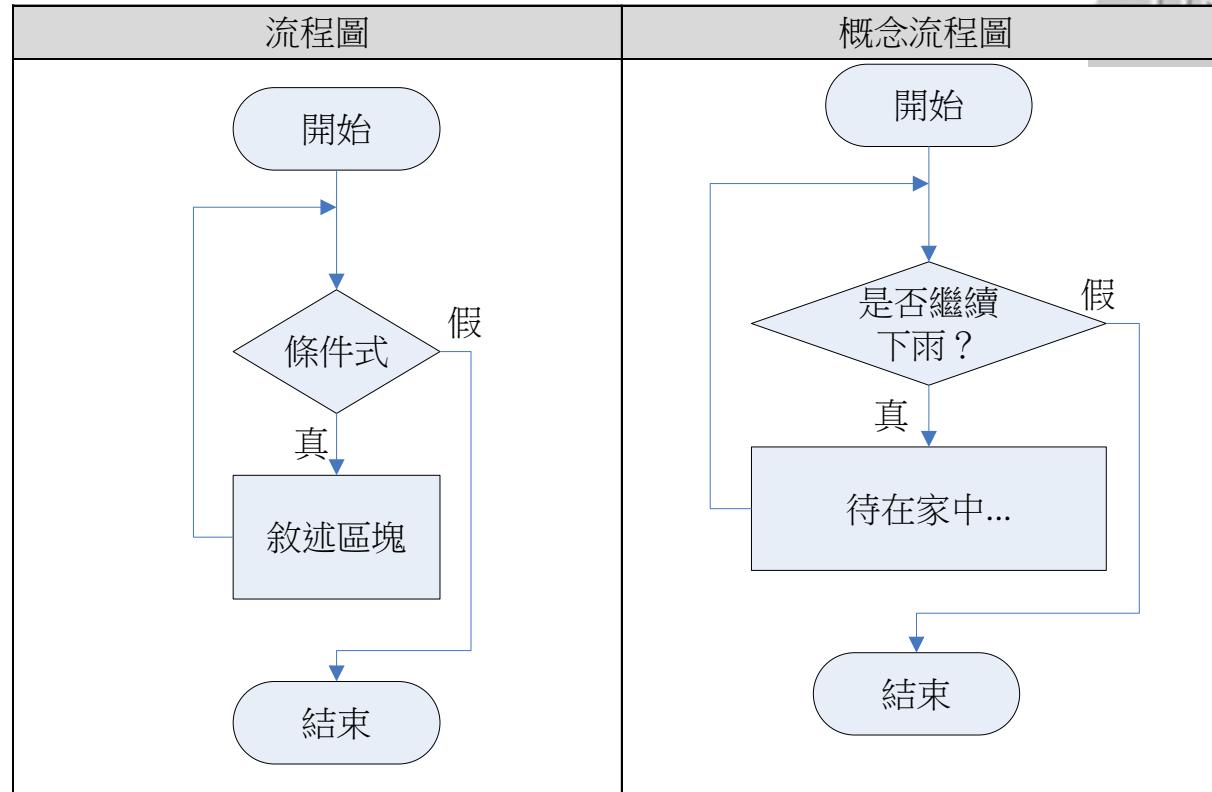
【語法】



【說明】while迴圈，當條件式「成立」時，才會反覆執行迴圈內的敘述區塊。

【注意】while敘述本身也可以形成巢狀迴圈(Nested Loop)或搭配for計數迴圈來構成巢狀迴圈。

【流程圖】



【實例1】利用while迴圈來求 $1+2+3+\dots+10$ 的程式

【解答】

程式檔案名稱		ch4-6-2-1A.java
01	public class ch4_6_2_1A	
02	{	
03	public static void main(String[] args)	
04	{ //宣告及初值設定	
05	int i = 1, Sum = 0;	
06	//處理	
07	while (i <= 10)	
08	{	
09	Sum = Sum + i;	
10	i = i + 1;	
11	}	
12	//輸出	
13	System.out.println("1+2+...+10=" + Sum);	
14	}	
15	}	

【執行結果】

1+2+...+10=55

雖然上面的例子，似乎無法明顯看出while迴圈的優點，但是，有些時候使用for迴圈並不容易描述我們想做的事情，例如如果我們想要知道2的多少次方才會大於1,000，由於事先並不知道到底是多少，所以很難設定for迴圈之中的變數值，勉強去用也會顯得很累贅，但是如果改用while迴圈，就會變得很簡單：

【實例2】求2的多少次方才會大於1,000呢？

【解答】



行號	ch4-6-2-1B.java
01	public class ch4_6_2_1B
02	{
03	public static void main(String[] args)
04	{ //宣告及初值設定
05	int i = 1, sum = 2;
06	//處理
07	while (sum < 1000)
08	{
09	sum = sum * 2;
10	i += 1;
11	}
12	//輸出
13	System.out.println ("2的" + i + "次方=" + sum);
14	}
15	}

【執行結果】

2的10次方=1024



4-6-2-2 後測試迴圈

【定義】先執行迴圈，再判斷條件式。

【作法】「判斷前」會先執行迴圈，當執行一次之後，再執行條件式判斷，不符合則跳出迴圈，但至少會執行一次迴圈。

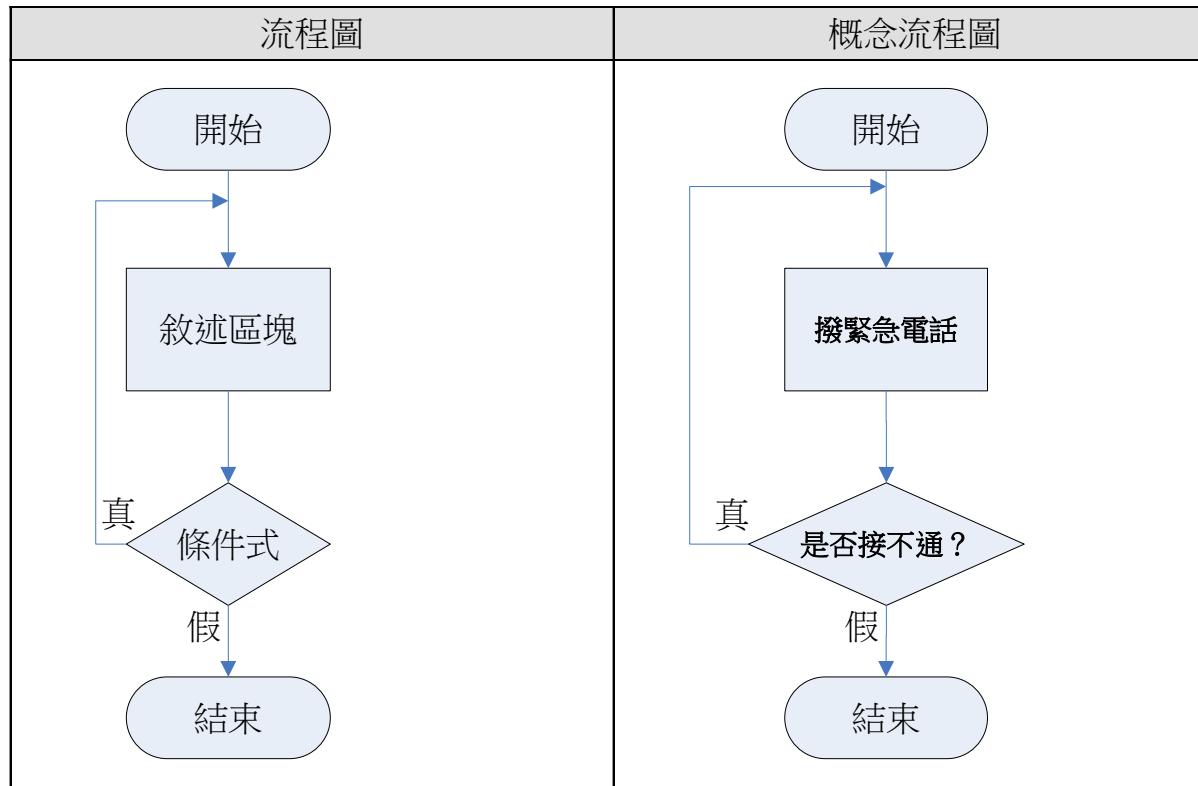
【語法】

```
do
{
    敘述區塊;
}
while(條件式);
```

【說明】當條件式為「不成立(False)」時，跳出迴圈。此迴圈會先執行迴圈一次之後，再執行條件式判斷，不符合則跳出迴圈，但至少會執行一次迴圈。也就是所謂的「先執行迴圈再判斷條件式」。

【注意】後測試迴圈的while(條件式)後面必須要加上分號(;), 而前測試迴圈的while則不需要。

【流程圖】



【實例】利用do/while來求 $1+2+3+\dots+10$ 的程式

【解答】

程式檔案名稱 ch4-6-2-2.java

```
01 public class ch4_6_2_2
02 {
03     public static void main(String[] args)
04     { //宣告及初值設定
05         int i = 1, Sum = 0;
06         //處理
07         do
08         {
09             Sum = Sum + i;
10             i = i + 1;
11         }while (i <= 10);
12         //輸出
13         System.out.println("1+2+...+10=" + Sum);
14     }
15 }
```

【執行結果】

1+2+...+10=55

4-7 break與continue敘述



break與continue敘述主要是應用在for與while迴圈當中，其中break敘述的目的就是用來中斷程式的執行動作。而continue敘述則是用來繼續執行某一動作。

4-7.1 break敘述



break敘述會使程式強迫跳離迴圈，繼續執行迴圈外下一個敘述，若其出現在巢狀迴圈內，則跳離該層迴圈，而break敘述在for與while迴圈中的比較如下：

for 迴圈	while 迴圈
<pre>for (計數變數=初值;終值條件;間隔值) { 程式區塊 1; break; ━━━━━━ 程式區塊 2; } 程式區塊 3;</pre>	<pre>while(條件式) { 程式區塊 1; break; ━━━━━━ 程式區塊 2; } 程式區塊 3;</pre>

【實例】 $1+2+3+\dots+10$ 的程式中加入break敘述的情況如下：

【解答】

程式檔案名稱		ch4_7_1A.java
01	public class ch4_7_1A	{ 02 { 03 public static void main(String[] args) 04 { //宣告 05 int i,Sum=0, <u>count</u> =0; 06 //處理 07 for(i=1;i<=10;i++) 08 { 09 if(i>5) break; 10 Sum=Sum+i; 11 } 12 //輸出 13 System.out.println("Sum=" + Sum); 14 } 15 } }

【說明】

1. 本迴圈的目的是就 $1+2+3+\dots+10$ ，但是當程式執行到行號09時，如果i值大於5時就會執行break動作，而跳出迴圈到行號13繼續執行。
2. 當i=6時，Sum之前已經從 $1+2+\dots+5$ 了，所以Sum=15。

【執行結果】

```
Sum=15
```

4-7.2 continue敘述



continue則是強迫程式跳到迴圈的起頭，當遇到其敘述時，停止執行迴圈主體，而到迴圈的開始處繼續執行，而continue敘述在for與while迴圈中的比較如下：

for 迴圈	while 迴圈
<pre>for (計數變數=初值;終值條件;間隔值) { 程式區塊 1; continue; _____ 程式區塊 2; } 程式區塊 3;</pre>	<pre>while(條件式) { 程式區塊 1; continue; _____ 程式區塊 2; } 程式區塊 3;</pre>

【實例2】 $1+2+3+\dots+10$ 的程式中加入continue敘述的情況如下：

【解答】

程式檔案名稱 | ch4_7_2A.java

```
01 public class ch4_7_2A
02 {
03     public static void main(String[] args)
04     { //宣告
05         int i,Sum=0;
06         //處理
07         for(i=1;i<=10;i++)
08         {
09             if(i>5) continue;
10             Sum=Sum+i;
11         }
12         //輸出
13         System.out.println("Sum=" + Sum);
14     }
15 }
```

說明：

1. 本迴圈的目的是就 $1+2+3+\dots+10$ ，但是當程式執行到行號09時，如果i值大於5時就會執行continue動作，而跳到for迴圈的條件式 $i \leq 10$ 繼續執行，而沒有執行到行號10的 $\text{Sum} = \text{Sum} + i$ 。
2. 當迴圈結束時，i值的已經累加到11，而Sum之前只計算從 $1+2+\dots+5$ ，所以 $\text{Sum} = 15$ 。

【課後評量】

1. 請問利用「結構化程式設計」的方式來撰寫程式，其優、缺點為何呢？
2. 請問利用【結構化程式設計】時，必須要遵守那些原則呢？
3. 在我們撰寫程式時，大部份都會先繪製流程圖，請問其目的為何呢？
4. 據你所知，繪製流程圖會有那些工具呢？至少寫出二種。
5. 請問「前測試迴圈」與「後測試迴圈」的差異點。
6. 請問「結構化程式設計」中，有那三種結構可以使用呢？並繪出其流程圖。
7. 請繪出求圓的面積(A)與周長(L)的流程圖。(註：輸入半徑R)
8. 請繪出求平均成績(Average)，並判斷是否及格的流程圖。

9. 請設計一個流程圖，可以讓使用者輸入一個整數，並判斷是否介於0~59之間，如果所輸入的值在範圍之內，則顯示正確，否則顯示不正確。



【分析】(1)輸入：一個整數為A

(2)處理：判斷是否介於0~59之間

(3)輸出：**①**成立顯示不及格

②不成立顯示及格

10. 請設計一個流程圖，可以讓使用者輸入三角形三邊長，求三角形的面積。

【分析】(1)輸入：三角形三邊長，分別為A,B,C

(2)處理：**①**判斷A,B,C是否全部都大於零

②並且任兩邊之和必須大於第三邊

③ $S = (A + B + C) / 2$

$$\text{面積公式} = (S(S-A)(S-B)(S-C))^{0.5}$$

(3)輸出：面積 真理大學 資訊工程學系

11. 請設計一個流程圖，可以讓使用者輸入A,B兩個值，並求 $AX+B=0$ 之解。

【分析】(1)輸入：兩個數值，分別為A,B

(2)處理：
①判斷A是否為零？若不為零，則 $X=-B/A$

②若A為零，再判斷B是否為零？

③若 $B=0$ ，則無限解

④若 $B \neq 0$ ，則無解

(3)輸出：X值或無限解或無解三種情況之一。

12. 請設計一個流程圖，可以讓使用者輸入三個數值，並求出最小值。

【分析】(1)輸入：三個數值，分別為A,B,C

(2)處理：
①先假設A為最小值，因此，令 $Min=A$

②判斷 $Min > B$ ，若成立則令 $Min=B$

③判斷 $Min > C$ ，若成立則令 $Min=C$

(3)輸出：顯示 Min ，即為最小值

13. 請設計一個流程圖，用來表達某公司的薪資計算方式，基本上，每小時100元，但是也有兩種不同的情況如下：

(1)工作時數超過50小時，加發10%工資

(2)工作時數少於50小時，加薪資300元。

試計算員工一個月的薪資。

【分析】(1)輸入：員工每月的工作時數hr

(2)處理：

①若 $hr > 50$ 小時，則薪資 $Salary = 100 * 50 + 100(hr - 50) * 1.1$

②若 $hr < 50$ 小時，則薪資 $Salary = 100 * hr - 300$

③若 $hr = 50$ 小時，則薪資 $Salary = 100 * hr$

(3)輸出：薪資Salary





14. 請設計一個程式，可以找出A,B兩數中的絕對值較大者。

【演算法】(1)輸入：二個數值，分別為A,B

(2)處理：
①如果 $A < 0$ ，則 $A = -A$

如果 $B < 0$ ，則 $B = -B$

②判斷 $A > B$ ，若成立則 $Max = A$

否則 $Max = B$

(3)輸出：顯示Max，即為最大值

15. 請設計一個程式，可以判斷是否為閏年。

【演算法】(1)輸入：一個西元年year

(2)處理：
①凡是能被4整除而不能被100整除者為閏年

②此外，可以被400整除者亦為閏年

(3)輸出：西元年是否為閏年

16. 請設計一個程式，可以讓使用者輸入三科成績，並求出最高分者。

<請利用巢狀迴圈>

【演算法】(1)輸入：三個數值，分別為A,B,C

(2)處理：
①如果 $A > B$ ，則繼續判斷A是否大於C，如果是的

話，A為最大值，否則C為最大值。

②如果 $A < B$ ，則繼續判斷B是否大於C，如果是的
話，B為最大值，否則C為最大值。

(3)輸出：顯示Max，即為最大值



17. 請設計一個程式，假設雄雄書局所賣的Java電腦書籍，其定價500元，有下面各種折扣方式：

1~5本書 不打折

6~10本書 照定價打9折

11~30本書 照定價打8.5折

31~50本書 照定價打8折

50本以上本書 照定價打7折

試設計一個程式，能輸入訂書量，計算出總售價。

【演算法】(1)輸入：訂書量

(2)處理：依照已知條件進行計算

(3)輸出：總售價



18. 請設計一個程式，例如有一個籠子中，雞兔共有10隻，而腳共有32隻，求雞兔各有幾隻。



【演算法】(1)輸入：雞兔總數與腳總數

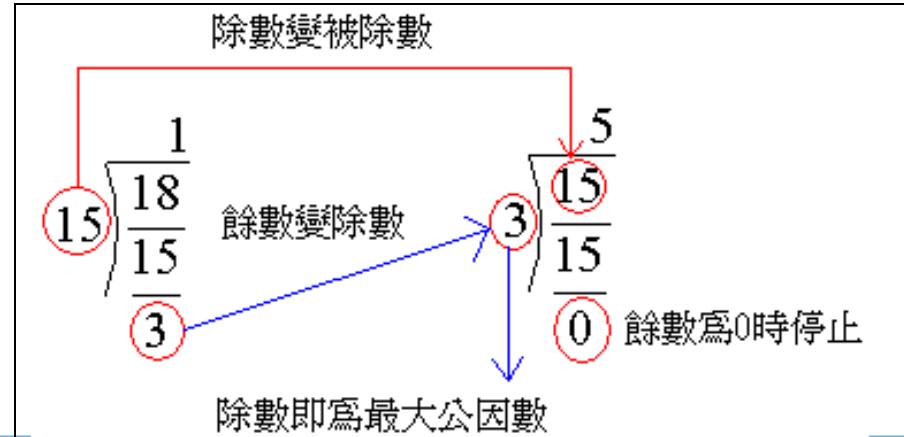
(2)處理：已知雞有2支腳而兔有4支腳

(3)輸出：雞兔各有幾隻

19. 請設計一個程式，可以求兩個數值的最大公因數與最小公倍數。

【演算法】(1)輸入：兩個數值A,B

(2)處理：利用「輾轉相除法」，求最大公因數。其概念如下所示：





說明：由上面可歸納三個重點

- ①每一回合之分母就是下一回合之分子。
- ②每一回合之餘數就是下一回合之分母。
- ③當餘數為零時，分母即為最大公因數。

(3)輸出：**①最大公因數**

②最小公倍數

20. 請設計一個程式，可以讓使用者輸入產生剪刀、石頭及布的程式。

【演算法】(1)輸入：0~3之任一個數值。

(2)處理：利用判斷式來判斷使用者輸入的數值。

- ①輸入1：產生剪刀
- ②輸入2：產生石頭
- ③輸入3：產生布
- ④輸入0：程式結束

(3)輸出：剪刀、石頭及布的任一個情況。

21. 請設計一個程式，假如某大學的資管系要甄選一位工讀生來當作系上的網頁程式設計的工作，其必須要同時符合以下的條件：

- ① 「國文」與「英文」之成績要60以上。
- ② 「程式設計」之成績要90(含)以上
- ③ 「資料庫系統」之成績要85以上
- ④ 「系統分析」要之成績85以上

請依照以上的條件來設計程式。

【演算法】(1)輸入：五科成績

(2)處理：判斷是否符合以上的已知條件

(3)輸出：①成立顯示符合

②不成立顯示不符合

22. 請設計一個程式，可以讓使用者輸入三角形三邊長，求三角形的面積。

【演算法】(1)輸入：三角形三邊長，分別為A,B,C

(2)處理：
①判斷A,B,C是否全部都大於零

②並且任兩邊之和必須大於第三邊

③ $S = (A + B + C) / 2$

面積公式 = $(S(S-A)(S-B)(S-C))^0.5$

(3)輸出：面積



23. 請設計一個程式，可以讓使用者輸入A,B兩個值，並求 $AX+B=0$ 之解。

【演算法】(1)輸入：兩個數值，分別為A,B

(2)處理：
①判斷A是否為零？若不為零，則 $X=-B/A$

②若A為零，再判斷B是否為零？

③若 $B=0$ ，則無限解

④若 $B \neq 0$ ，則無解

(3)輸出：X值或無限解或無解三種情況之一。





24. 請設計一個程式，用來計算某公司的薪資計算方式，基本上，每小時100元，但是也有兩種不同的情況如下：

(1)工作時數超過50小時，加發10%工資

(2)工作時數少於50小時，加薪資300元。

試計算員工一個月的薪資。

【演算法】(1)輸入：員工每月的工作時數hr

(2)處理：

①若 $hr > 50$ 小時，則薪資 $Salary = 100 * 50 + 100(hr - 50) * 1.1$

②若 $hr < 50$ 小時，則薪資 $Salary = 100 * hr - 300$

③若 $hr = 50$ 小時，則薪資 $Salary = 100 * hr$

(3)輸出：薪資Salary

25. 請設計一個程式，假設有某一家公司的員工職位分為五個等級，每月薪資計算方式如下：

職等	1	2	3	4	5
每日薪資	2000	1500	1000	800	500

【演算法】(1)輸入：員工的職等State及工作天數days

(2)處理：

依照職等State來計算總薪資(Total)=每日薪資(salary)*
天數days

(3)輸出：總薪資(Total)



26. 請設計一個程式，可以輸入三角形的三個邊長，分別為A,B,C，並且判斷此三角形為何種三角形。

【條件】①直角三角形為： $A^2=B^2+C^2$

②純角三角形為： $A^2 > B^2+C^2$

③銳角三角形為： $A^2 < B^2+C^2$

27. 請設計一個程式，可以在輸入一個正整數後，判斷此數是否為2或3的倍數

【演算法】(1)輸入：一個正整數N

(2)處理：①N被2及3整除

② N被2，但沒有被3整除

③ N不被2整除，但被3整除

④ N不被2及3整除

(3)輸出：是否為2或3的倍數

28. 請設計一個程式，可以在輸入兩個正整數後，能判斷此二數是否恰為一奇一偶。

【演算法】(1)輸入：兩個正整數A,B

(2)處理：①判斷 $(A+B)$ 被2除之後的餘數是否為1

②如果成立，則代表兩數恰為一奇一偶

如果不成立，則代表兩數並非為一奇一偶

(3)輸出：二數是否恰為一奇一偶

29. 請設計一個程式，可以在輸入一個數字後，能判斷此數是否為一個兩位數之整數。

【演算法】(1)輸入：一個正整數N

(2)處理：判斷N值的範圍是否介於10~99或-10 ~ -99之間

(3)輸出：是否為一個兩位數之整數

30. 王媽媽第一天給小明10元，第二天給20元，第三天給30元，請問小明要儲存到5000元需要多少天？

31. $1+4+7+10+\dots+X$ 之總和不超過200，請設計一個程式以求出X的最大值。

32. 請設計一個程式可以求出1~100之間所有偶數個數及和。

33. 請設計一個程式可以求出1~100之間所有奇數個數及和。

34. 請設計一個程式可以求出1~100之間所有偶數平方和。

35. 請設計 $1+1/2+1/3+\dots+1/N$ 之和。

36. 請設計 $1+1/3+1/5+\dots+1/N$ 之和。

37. 請設計 $1+1/2+1/4+\dots+1/N$ 之和。

38. 請設計 $1+1/2+1/4+1/8+1/16\dots+1/N$ 之和。

39. 請撰寫一個程式來計算以下的數學式：

$$\text{求 } \frac{1*2}{3*4} + \frac{5*6}{7*8} + \dots + \frac{101*102}{103*104} \text{ 之和}$$

40. 請設計 $0 \sim 10!$ 的計算過程。如下圖所示：



```
0!=1  
1!=1=1  
2!=1*2=2  
3!=1*2*3=6  
4!=1*2*3*4=24  
5!=1*2*3*4*5=120  
6!=1*2*3*4*5*6=720  
7!=1*2*3*4*5*6*7=5040  
8!=1*2*3*4*5*6*7*8=40320  
9!=1*2*3*4*5*6*7*8*9=362880  
10!=1*2*3*4*5*6*7*8*9*10=3628800
```

41. 輸入一正整數N，求 $1-2+3-4+\dots+(-1)^n \cdot n = ?$

42. 輸入一正整數N，求 $1-\frac{1}{2}+\frac{1}{3}-\frac{1}{4}+\dots+(-1)^{n-1} \cdot \frac{1}{n} = ?$

43. 輸入一正整數N，列出此數之所有因數。

44.請輸出正三角的圖。

```
$  
$$  
$$$  
$$$$  
$$$$$
```



45.請輸出倒三角的圖。

```
$$$$$  
$$$$  
$$$  
$$  
$
```

46. 請輸出錢字號堆疊成 K 字型的圖。

```
$$$$$  
$$$$  
$$$  
$$  
$  
$$  
$$$  
$$$$  
$$$$$
```



47. 請設計下列的數字三角形程式。

```
1  
12  
123  
1234  
12345  
123456
```